6-2017

Fusing mobile, wearable and infrastructure sensing for immersive daily lifestyle analytics

Sougata SEN
Singapore Management University, sougata.sen.2012@phdis.smu.edu.sg

Fusing Mobile, Wearable and Infrastructure Sensing for Immersive Daily Lifestyle Analytics

SOUGATA SEN

SINGAPORE MANAGEMENT UNIVERSITY
2017

Fusing Mobile, Wearable and Infrastructure Sensing for Immersive Daily Lifestyle Analytics

by

Sougata Sen

Submitted to School of Information Systems in partial fulfillment of the requirements for the Degree of Doctor of Philosophy in Information Systems

Dissertation Committee:

Archan MISRA (Supervisor / Chair)
Associate Professor of Information Systems
Singapore Management University

Rajesh Krishna BALAN (Co-supervisor)
Associate Professor of Information Systems
Singapore Management University

Youngki LEE
Assistant Professor of Information Systems
Singapore Management University

Andrew T. CAMPBELL
Professor of Computer Science
Dartmouth College

Singapore Management University
2017

Fusing Mobile, Wearable and Infrastructure Sensing for Immersive Daily Lifestyle Analytics

Sougata Sen

Abstract

With the prevalence of sensors in public infrastructure as well as in personal devices, exploitation of data from these sensors to monitor and profile basic activities (e.g., locomotive states such as walking, and gestural actions such as smoking) has gained popularity. Basic activities identified by these sensors will drive the next generation of lifestyle monitoring applications and services. To provide more advanced and personalized services, these next-generation systems will need to capture and understand increasingly finer-grained details of various common daily life activities.

In this dissertation, I demonstrate the possibility of building systems using off-the-shelf devices, that not only identify activities, but also provide fine-grained details about an individual's lifestyle, *using a combination of multiple sensing modes*. These systems utilise sensor data from personal as well as infrastructure devices to unobtrusively monitor the daily life activities. In this dissertation, I have used eating and shopping as two examples of daily life activities and have shown the possibility to monitor fine-grained details of these activities. Additionally, I have explored the possibility of utilising the sensor data to identify the cognitive state of an individual performing a daily life activity.

I first investigate the possibility of using multiple sensor classes on wearable devices to capture novel context about common gesture-driven activities. More specifically, I describe *Annapurna*, a system which utilises the inertial and image sensors in a single device to identify fine-grained details of the eating activity. *Annapurna* utilises data from the inertial sensors of a smartwatch efficiently to determine when a person is eating. The inertial sensors opportunistically trigger the smartwatch's camera to capture images of the food consumed, which is used in building a food journal. *Annapurna* has been subjected to multiple user studies and we found that

the system can capture finer details about the eating activity – images of the food consumed, with false-positive & false-negative rates of 6.5% & 3.3% respectively.

I next investigate the potential of combining sensing data from not just multiple personal devices, but also by using inexpensive ambient sensors/IoT platforms. More specifically, I describe I^4S, a system utilises multiple sensor classes in multiple devices to identify fine-grained in-store activities of an individual shopper. The goal of I^4S is to identify all the items that a customer in a retail store interacts with. I^4S utilises the inertial sensor data from the smartwatch to identify the picking gesture as well as the shelf from where an item is picked. It utilises the BLE scan information from the customer's smartphone to identify the rack from where the item is picked. By analysing the data collected through a user study involving 31 users, we found that we could identify pick gestures with a precision of over 92%, the rack where the pick occurred with an accuracy of over 86% and identify the position within a 1 meter wide rack with an accuracy of over 92%.

Finally, I explore the possibility of using such finer-grained capture of an individual's physical activities to infer higher-level, cognitive characteristics associated with such daily life activities. As an exemplar, I describe *CROSDAC*, a technique to identify the cognitive state and behavior of an individual during the shopping activity. To determine the shopper's behavior, *CROSDAC* analyses the shopper's trajectory in a store as well as the physical activities performed by the shopper. Using an unsupervised approach, *CROSDAC* first discovers clusters (i.e., implicitly uncovering distinct shopping styles) from limited training data, and then builds a cluster-specific, but person-independent, classifier from the modest amount of training data available. Using data from two studies involving 52 users conducted in two diverse locations, we found that it is indeed possible to identify the cognitive state of the shoppers through the *CROSDAC* approach.

Through these three systems and techniques, in this dissertation I demonstrate the possibility of utilising data from sensors embedded in one or more off-the-shelf devices to determine fine-grained insights about an individual's lifestyle.

Table of Contents

1 Introduction **1**

1.1 Overview 1

1.2 Challenges 5

1.3 Thesis Statement 11

1.4 Research Contribution 13

1.5 Project Contributions 18

1.6 Dissertation Roadmap 19

2 Literature Review **22**

2.1 Activity Recognition 23

2.1.1 Addressing Challenges in Activity Recognition 24

2.1.2 Identifying and Monitoring Specific Daily Life Activities . . 25

2.2 Eating Activity Recognition 28

2.3 Shopping Activity Recognition 33

2.4 Behavior Recognition 37

2.4.1 Understanding the Shopping Behavior 37

2.4.2 Automatically Determining Emotions and Behaviors 38

3 Monitoring Activities of Daily Living (ADL) **41**

3.1 Identifying and Understanding ADL 41

3.2 Motivating Scenarios 43

3.2.1 Scenario 1 43

3.2.2 Scenario 2 . . . 44
3.2.3 Scenario 3 . . . 46
3.2.4 Other Scenarios . . . 47
3.3 ADL Monitoring Systems and Techniques . . . 48
3.3.1 *Annapurna*: Automated Food Journaling . . . 48
3.3.2 I^4S: Identifying In-store Interactions . . . 50
3.3.3 *CROSDAC*: Understanding Shopping Behavior . . . 51

4 Automated Food Journaling 52
4.1 Need for Automated Food Journaling . . . 53
4.2 System Overview . . . 54
4.2.1 Design Goals . . . 55
4.2.2 Overview . . . 55
4.3 Design Choices . . . 57
4.3.1 Micro Studies and Observations . . . 58
4.3.2 Choices That Did Not Work . . . 68
4.3.3 In-the-Wild Studies . . . 69
4.4 Methodology & Results . . . 72
4.4.1 Detecting Eating Gestures . . . 72
4.4.2 Capturing Food Images . . . 80
4.5 *Annapurna* Application . . . 86
4.5.1 Watch and Phone Modules . . . 86
4.5.2 Server Module and Parameter Choices . . . 87
4.5.3 User Feedback and Opinions . . . 88
4.6 Discussion . . . 89
4.7 Summary . . . 92

5 Identifying Fine-Grained In-store Shopper Interactions 94
5.1 Necessity of Capturing In-Store Interactions . . . 95
5.2 System Overview . . . 100

5.3 Design Choices . . . 103
5.3.1 Dataset . . . 104
5.3.2 Inertial Sensor Analysis for Gesture Recognition . . . 106
5.3.3 Bluetooth Low Energy (BLE) Analysis . . . 107
5.3.4 Magnetic Field Sensor Analysis . . . 112
5.4 Methodology . . . 112
5.4.1 Pick Gesture Detection . . . 113
5.4.2 Rack Level Pick Location Identification . . . 115
5.4.3 Shelf Level Pick Location Identification . . . 117
5.5 Results . . . 120
5.5.1 Pick Gesture Identification . . . 120
5.5.2 Rack Level Location Identification . . . 123
5.5.3 Shelf Level Location Detection . . . 126
5.5.4 Summary of I^4S Approach . . . 128
5.6 Discussion . . . 128
5.7 Summary . . . 131

6 Understanding Individual's Behaviour 132
6.1 Necessity of Identifying Shopping Behaviour . . . 133
6.2 System Overview . . . 135
6.3 Design Choices . . . 140
6.3.1 Datasets . . . 141
6.3.2 Determining Number of Shopping Styles . . . 146
6.4 Methodology . . . 148
6.4.1 Feature Vectors & Classification . . . 149
6.5 Results . . . 150
6.5.1 Study 1: Food Court . . . 151
6.5.2 Study 2: University Gift Shop . . . 154
6.6 Discussion . . . 158

6.7 Summary 160

7 Discussion and Future Directions **162**

7.1 Additional Uses of Gesture-Triggered Image Capturing 164

7.1.1 In-Store Interaction Monitoring 164

7.2 Short Term Plan 172

7.2.1 Automated Food Journaling 172

7.2.2 In-store Interaction Identification 173

7.3 Longer Term Research 174

8 Conclusion **178**

8.1 System and Technique Summary 178

8.2 Closing Remarks 180

List of Figures

3.1 Breakdown of Global Wearable Sales 42

4.1 Smartwatches with Embedded Cameras 54
4.2 System Overview of *Annapurna* 56
4.3 Capturing Food Images vs. Smartwatch Position 60
4.4 Frames Extracted from the Video During a Complete Eating Gesture 61
4.5 Frames Captured in Preview Mode During a Complete Eating Gesture 62
4.6 Sample Images Classified as Usable Images by the Two Annotators 63
4.7 Sample Images Classified as Not-Usable Images by the Two Annotators 63
4.8 Evaluation of Two Strategies to Capture Food Images 65
4.9 Image Label Prediction Using a Commercial Image Recognition System for the Not-Useful Images. 65
4.10 Images Captured in Preview and Video Mode. (Scale 1:2) 67
4.11 Eating Period Recognition Approach 72
4.12 Variation of Accuracy as Training Data Size is Varied 74
4.13 Variation of Accuracy as Number of Users is Varied 75
4.14 Error Rates for Different Cost Parameters 79
4.15 Power Consumption by Various Components 81
4.16 Output of Edge Detection 82
4.17 Bounding Box Extrapolation to Determine Maximum Area 83
4.18 Algorithm for Ranking Images 84

4.19 Images with Human Faces Detected 85
4.20 Snapshot of *Annapurna* Portal Shown to a User 87
4.21 Number of Images to be Displayed in *Annapurna* Web Portal. . . . 88

5.1 Store Layout and Distinct Terms 97
5.2 Pictures to Estimate In-Store Item Density 98
5.3 Overview of I^4S System Working 100
5.4 Overview of the I^4S System with Smartwatch and Smartphone . . . 103
5.5 In-Lab Data Collection Location 104
5.6 Shop Where Data Collection was Performed 104
5.7 Ground Truth Data Collection Application Screenshot 106
5.8 Smartwatch's Accelerometer's Data Variation for a Shopping Episode 106
5.9 Orientation of Different Axis when the Watch is Worn on the Hand . 106
5.10 Accelerometer Variation for Picks from a Lower Shelf Under Controlled Conditions. 107
5.11 Accelerometer Variation for Picks from Top Shelf Under Controlled Conditions. 107
5.12 Difference in Number of Beacons Heard by Phone and Watch 108
5.13 Episode-wise Ratio of Beacons Heard by Phone and Watch 109
5.14 Difference in Received Signal Strength Indicator (RSSI) Between Phone and Watch for an Episode 109
5.15 Timeline Showing When a Beacon was Heard by the Device 110
5.16 Variation of Magnetometer Readings Inside the Store 112
5.17 Trellis for Viterbi Smoothing 117
5.18 Variation of Accuracy/Precision/Recall of 10 Fold Cross Validation for Different Cost Parameter Settings for Picking Being Misclassified 121
5.19 Variation of Accuracy Across Users for Leave-One-User-Out Cross Validation 122
5.20 Variation of Leave-One-Pick-Out Accuracy for Varied Beacon Count 124

5.21 Variation of Prediction Accuracy Across Different Racks 125
5.22 Surface Chart Showing Zone Wise Location Prediction Accuracy . . 126

6.1 *SHOP* Overall Architecture 135
6.2 Steps in *CROSDAC* Classification 136
6.3 Schematic Layout of Mall 142
6.4 Schematic Layout of University Gift Shop 145
6.5 Study 1: Effect of Cluster Size on Prediction Accuracy 147
6.6 Study 2: Effect of Cluster Size on Prediction Accuracy 148
6.7 Study 1: Features with the Highest Information Gain 153
6.8 Study 2: Features with the Highest Information Gain 155
6.9 Average Performance (20-Runs) of *CROSDAC* for Different Training Data Size 157

7.1 Architecture for a Single Device Item Identification System 165
7.2 Camera View With Image Captured While an Item is Being Picked . 167
7.3 Images Extracted from the Video While Person is Picking an Item . 168
7.4 Probability of Capturing Image of Item Being Picked 169

List of Tables

1.1 Project Contributions – *Annapurna* 18

1.2 Project Contributions – I^4S 18

1.3 Project Contribution – *CROSDAC* 19

2.1 Comparison of Various Dailly Life Activity Monitoring Systems . . 26

2.2 Comparison of Various Eating Activity Recognition Techniques . . 29

2.3 Comparison of Various Shopping Activity Recognition Techniques 35

3.1 List of Devices Used in the Studies 51

4.1 Devices Used in Realising *Annapurna* 53

4.2 Key Results from Micro Studies 58

4.3 Details of In-the-Wild Studies for *Annapurna* 70

4.4 System Performance for Each Individual Particating in the In-the-Wild Study:3 71

4.5 Accuracy in Identifying Eating Gestures 73

4.6 Gesture Prediction Error (%) vs. (Window Size, Threshold) 77

4.7 Differences in Gesture Prediction Error (%) Between Rice & Noodles 77

4.8 Sensor Data Based Gesture Count Determination 78

4.9 Effectiveness of Image Filtering 85

4.10 User Feedback for the Overall *Annapurna* System 89

5.1 Devices Used in I^4S 95

5.2 Summary of Dataset Collected In-Store 104

5.3 Features Extracted from Inertial Sensors 113
5.4 Features Extracted from Game Rotation Vector Sensor 119
5.5 Accuracy (Precision/Recall) in Identifying Picking Gesture 121
5.6 Precision and Recall in Identifying Picking Gesture in a Person Independent Setting with Varying Smoothing Window Length 123
5.7 Variation of Accuracy when Fingerprint is Generated Based on Number of Times Beacon is Heard 124
5.8 Confusion Matrix for Zone in Shelf Identification 127
5.9 Summary of the Performance of Various Components of I^4S 128

6.1 Various Approaches for Crowd-Scale Shopping Behavior Prediction 139
6.2 Summary of the Studies Conducted to Understand Shopper's Behavior 141
6.3 Study 1: Classification Accuracy for Sensor Data and Ground Truth 151
6.4 Study 2: Performance of Different Approaches in the University Gift Shop 154

Acknowledgements

I am indebted to several individuals who have been instrumental in ensuring that I complete the Ph.D. journey.

First of all, I would like to thank my advisor, Associate Professor Archan Misra for his support and guidance throughout the Ph.D. journey. I am extremely fortunate to be advised by him and could not have imagined having a better mentor for my Ph.D. His passion for research, enthusiasm for work and being a perfectionist has always been an inspiration for me. I have learnt to research systematically from him. Other than the professional advise and support, I have received ample support from him on a personal front. He is one of the kindest and wittiest person that I know. I couldn't have completed the Ph.D. journey without his support and mentoring and words are not enough to express my gratitude.

I would also like to express my gratitude to my stellar dissertation committee: my co-advisor – Associate Professor Rajesh Krishna Balan, Assistant Professor Youngki Lee and Professor Andrew Thomas Campbell. My special thanks to Rajesh, who helped and guided me at every stage of the Ph.D. journey. I have received invaluable reviews and guidance from him for every project. I thank Youngki, with whom I have worked on several projects. His keenness in discussing minute details about the projects has helped me in improving the systems that I have developed. I thank Andrew, who has provided extremely valuable feedback for the dissertation. I have always been inspired by his research and his detailed feedback for the dissertation has helped in improving its standard.

I have been lucky to have several collaborators and mentors during various pha-

ses of my Ph.D. I will like to specially thank Vigneshwaran Subbaraju for every collaboration. I have learnt a lot from him. I will also like to thank Dipanjan Chakraborty, Dipyaman Banerjee, Tianli Mo, Lipyeow Lim, Vijay Srinivasan, Kiran Rachuri and Abhishek Mukherji for their collaborations, guidance and mentoring.

I would like to thank Pei Huan and Ong Chew Hong for monitoring my academic progress closely. I also thank Huang Sipei, Luar Shu Hui, Kazae Quek and Jonathan Wang for providing assistance whenever I needed. A special thank you to Kazae for helping me in beautifying my presentation slides and figures in my papers. I thank LiveLabs, and MOE funding for providing me with travel grants to attend conferences and present my papers. I also thank LARC for providing me the opportunity of spending a year at Carnegie Mellon University as a research scholar.

I have been blessed with a lot of friends, who have been instrumental in keeping me sane during the Ph.D. journey. I will like to thank all my lab-mates and colleagues in Livelabs and SIS, with whom I have been associated with at various phases of my journey. Kartik Muralidharan, Luong Trung Tuan, Rijurekha Sen, Tan Kiat Wee, Joseph Chan, Kasthuri Jayarajah, Meeralakshmi Radhakrishnan, Jeena Sebastian, Swetha Gottipati, Huynh Nguyen, Amit, Hai Le Gia, Thanh Chau, Swapna Gottipati, and Payas Gupta, all of you have helped in making the journey easier.

Finally, I sincerely thank my family and friends for their support and encouragement. I am grateful to my parents for the unconditional love, sacrifices and for supporting every decision I have taken. I hope the little boy who hated going to school has made them proud. I thank my brothers Subhagata and Sourav for making several life choices and sacrifices – knowingly or unknowingly, without which I couldn't have attended grad school. I thank my in-laws for being patient and encouraging during this journey. A big shout-out to my friends Avinash Bahirvani and Himangshu Dutta for being there whenever I needed them. My wife, Komal has been a constant support throughout the Ph.D. journey. I thank her for being patient (even during paper deadlines) and making me believe that I would sail through, even when I was unreasonable, unaccommodating, unsocial,....

To

Komal, for always being patient, understanding, and supportive

&

Ma and Baba, for their unconditional love and encouragement

List of Publications

Conference/Workshop Papers

Sougata Sen, Vigneshwaran Subbaraju, Archan Misra, Rajesh Balan and Youngki Lee. Experiences in Building a Real-World Eating Recogniser. *In Proceedings of the 4th workshop on Workshop on Physical Analytics*, Niagara Falls, NY, 2017 DOI: http://dx.doi.org/10.1145/3092305.3092306

Sougata Sen, Kiran Rachuri, Abhishek Mukherji and Archan Misra. Did you take a break today? Detecting playing foosball using your smartwatch. *In Proceedings of IEEE International Conference on Pervasive Computing and Communication Workshops (PerCom Workshops)*, Sydney, NSW, 2016, pp. 1-6.
DOI: 10.1109/PERCOMW.2016.7457165

Meera Radhakrishnan, **Sougata Sen**, Vigneshwaran Subbaraju, Archan Misra and Rajesh Balan. IoT+Small Data: Transforming in-store shopping analytics & services. *In Proceedings of 8th International Conference on Communication Systems and Networks (COMSNETS)*, Bangalore, 2016, pp. 1-6.
DOI: 10.1109/COMSNETS.2016.7439946

Sougata Sen. Pervasive physical analytics using multi-modal sensing. *In Proceedings of 8th International Conference on Communication Systems and Networks (COMSNETS)*, Bangalore, 2016, pp. 1-2. DOI: 10.1109/COMSNETS.2016.7439998

Vigneshwaran Subbaraju, **Sougata Sen**, Archan Misra, Satyadip Chakraborti and Rajesh Balan. Using infrastructure-provided context filters for efficient fine-grained activity sensing. *In Proceedings of 2015 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, St. Louis, MO, 2015, pp. 87-94. DOI: 10.1109/PERCOM.2015.7146513

Sougata Sen. Opportunities and challenges in multi-modal sensing for regular lifestyle tracking. *In Proceedings of 2015 IEEE International Conference on Pervasive Computing and Communication Workshops (PerCom Workshops)*, St. Louis, MO, 2015, pp. 225-227. DOI: 10.1109/PERCOMW.2015.7134030

Sougata Sen, Vigneshwaran Subbaraju, Archan Misra, Rajesh Balan and Youngki Lee. The case for smartwatch-based diet monitoring. *In Proceedings of the IEEE International Conference on Pervasive Computing and Communication Workshops (PerCom Workshops)*. St. Louis, MO, 2015, pp. 585-590.
DOI: 10.1109/PERCOMW.2015.7134103

Sougata Sen, Dipanjan Chakraborty, Vigneshwaran Subbaraju, Dipyaman Banerjee, Archan Misra, Nilanjan Banerjee, and Sumit Mittal. Accommodating user diversity for in-store shopping behavior recognition. *In Proceedings of the 2014 ACM International Symposium on Wearable Computers (ISWC '14)*. ACM, New York, NY, USA, 11-14. DOI=http://dx.doi.org/10.1145/2634317.2634338

Tianli Mo, **Sougata Sen**, Lipyeow Lim, Archan Misra, Rajesh Balan and Youngki Lee. Cloud-Based Query Evaluation for Energy-Efficient Mobile Sensing. *In Proceedings of IEEE 15th International Conference on Mobile Data Management*. Brisbane, QLD, 2014, pp. 221-224. DOI: 10.1109/MDM.2014.33

Sougata Sen and Kartik Muralidharan. Putting pressure on mobile authentication. *In Proceedings of Seventh International Conference on Mobile Computing and Ubiquitous Networking (ICMU)*, Singapore, 2014, pp. 56-61.
DOI: 10.1109/ICMU.2014.6799058

Sougata Sen, Archan Misra, Rajesh Balan, and Lipyeow Lim. The case for cloud-enabled mobile sensing services. *In Proceedings of the first edition of the MCC workshop on Mobile cloud computing (MCC '12)*. ACM, New York, NY, USA, 53-58. DOI=http://dx.doi.org/10.1145/2342509.2342521

Journal Papers

Tianli Mo, Lipyeow Lim, **Sougata Sen**, Archan Misra, Rajesh Krishna Balan, Youngki Lee, Cloud-based query evaluation for energy-efficient mobile sensing, Pervasive and Mobile Computing. *Pervasive and Mobile Computing* ISSN 1574-1192, http://dx.doi.org/10.1016/j.pmcj.2016.12.005.

Demos/Posters

Sougata Sen, Vigneshwaran Subbaraju, Archan Misra, Youngki Lee, and Rajesh Krishna Balan. 2016. Demo: Smartwatch based Food Diary & Eating Analytics. *In Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services Companion (MobiSys '16 Companion)*. ACM, New York, NY, USA, 118-118. DOI: http://dx.doi.org/10.1145/2938559.2938569

Meeralakshmi Radhakrishnan, Sharanya Eswaran, **Sougata Sen**, Vigneswaran Subbaraju, Archan Misra, and Rajesh Krishna Balan. Demo: Smartwatch based Shopping Gesture Recognition. *In Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services Companion (MobiSys '16 Companion)*. ACM, New York, NY, USA, 115-115.
DOI: http://dx.doi.org/10.1145/2938559.2938572

Sougata Sen, Dipanjan Chakraborty, Dipyaman Banerjee, Archan Misra, Nilanjan Banerjee, Vigneshwaran Subbaraju, and Sumit Mittal. Poster: SHOP: Store Habits Of People *15th Workshop on Mobile Computing Systems and Applications (HotMobile '14)*

Chapter 1

Introduction

1.1 Overview

Over the last few years, recognising simple locomotive activities performed by individuals through the course of a day (e.g. walking, standing, sitting, etc.) using mobile and wearable devices has become common [16, 56, 59, 81, 123, 151, 159]. The ability to derive these simple activities has now led researchers to explore the possibility of using sensor data from personal devices to identify and monitor more complex activities, where a complex activity might be one that occurs naturally as a result of an individual's daily lifestyle (e.g., the capture of daily eating behavior [88, 133, 145] or the study of sleeping patterns and phases [25, 87, 96]) or might refer to specialized behavior (e.g., the tracking of abnormalities in gait [84, 98] or monitoring lower limb exercises performed at a gym [167]). However, many of these monitoring systems are either designed for specific environments or derive only a part of the activity's context (e.g. determining eating, without identifying what was eaten). An open and exciting question is whether fine-grained details of such complex activities can be reliably and accurately inferred by by utilizing the diversity of sensors present in either a single or multiple off-the-shelf devices, especially given (a) the wide variation in the way different individuals (and the same individual on different occasions) perform such activities, and (b) the variation in

the capability of an individual sensor in a device as well as a device to monitor a specific context.

This dissertation explores the possibility of using one or more off-the-shelf devices to obtain insights about an individual's daily lifestyle activities, at either finer granularity than previously possible or for attributes that have previously not been easily monitored. The dissertation does not develop novel mobile or wearable based activity recognition techniques, but explores the possibility of using existing approaches along with IoT based techniques to monitor fine-grained details of interesting daily life activities. For this exploration, it uses two commonplace lifestyle activities, *eating* and *shopping* as exemplars. Eating and shopping are of particular interest, as the ability to unobtrusively monitor these activities of an individual have high value for a variety of future applications and services, especially in the areas of wellness and retail. According to medical literature, these two activities fall under the broader umbrella of Activities of Daily Living (ADL), where eating is an essential ADL, while shopping falls under the category of Instrumental ADL (IADL) [40] (The importance of monitoring IADL tasks is described in [38]). Moreover, these two activities serve as useful exemplars of two classes of such ADLs, one which can occur anywhere (individuals can eat at a wide variety of places) and another which occurs only at specific locations (within stores) and can thus take advantage of specific infrastructural instrumentation. In this dissertation, I determine the extent to which personal devices are sufficient for monitoring such activities, and the additional advantages that can arise from the use of infrastructure sensing in specifically instrumented locations.

Over the years, for understandable reasons, monitoring daily life activities has been of significant interest in the medical domain, where approaches such as external observations or maintaining a self-reported diary or blog [126] or instrumenting subjects with sensors [106] has been investigated. However, I believe that if the monitoring of ADLs becomes unobtrusive and unnoticeable, it can have significant impact beyond the medical domain. Naturally, sensor data from mobile, wearable or

infrastructure devices used either individually or collaboratively can be instrumental in this transition. In this dissertation, I explore various innovative techniques, which in addition to utilising data from mobile and wearable devices, also utilises sensor data from infrastructure sensors to analyse various interesting lifestyle activities.

Various ADL monitoring systems utilising either one or multiple devices amongst smartphones, wearables or infrastructure sensors have been proposed by researchers. The evolution of these systems and techniques reveals a progressive change, not only in the devices or techniques used, but also in the fine-grained ADL monitoring goals that researchers are attempting to achieve. Infrastructure based ADL monitoring techniques (e.g. Tapia et. al. [143]) were proposed even before it was established that smartphones could be used for activity monitoring. Once it was established that smartphones could be used for activity recognition, researchers increasingly focused on using smartphones for user context monitoring, either through a single phone (e.g. - CenceMe [86] and SurroundSense [8]) or through collaborative monitoring using multiple smartphones (e.g. - Darwin Phones [85] and CoMon [68]). The next wave of interesting applications arrived once wearables (smartwatch, smartglass, fitness bands etc.) became popular (e.g. RisQ [104] tracks a natural gesture (smoking) through a smartwatch, while E-Gesture [105] tracks application-specific custom arm gestures). Researchers have additionally explored techniques for collaboratively monitoring context using heterogeneous devices (e.g. ThirdEye [122]), which explored how a smartglass could be used to track a user's visual exploration of products in retail environments. Similar to RisQ [104], this dissertation demonstrates the possibility of utilising sensor data from a fixed-positioned wearable device (smartwatch) to identify natural gestures. However, in addition to identifying the gestures, this dissertation describes several techniques to infer finer insights of the lifestyle activity (e.g., the image of the food being consumed while eating, or the shelf-level location from where a consumer picks products while shopping) once the gesture corresponding to the activity is identified. These techniques either utilise multiple sensor classes within the same device for finer

activity insights or utilise sensors on one device as a trigger for capturing or processing sensor data on another device. In this dissertation, I have borrowed several existing mobile/wearable based activity recognition techniques and have used these techniques along with IoT based context recognition approaches for fine-grained monitoring of daily lifestyle activities.

The evolution of multi-sensor and multi-device fine grained lifestyle monitoring has been possible because of a number of important technological advances. Some of these include:

1. Research in the field of connected sensors as well as wireless sensor networks [49] has enabled various localized (e.g., in homes and shops) and city-wide infrastructure based IoT deployment with the dream of realising smart cities [137]. In addition to deriving home, shop or city level analytics, the sensor data from these devices can be used for personal level context identification (e.g. utilising Bluetooth Low-Energy (BLE) beacons deployed in a food court as a location trigger for monitoring an individual's eating activity).

2. The increasing diversity of sensors embedded in smartphones and wearables has established the possibility of monitoring fine grained contexts of activities performed by an individual without completely depending on external sources.

3. The possibility of programming the smart devices and even moving part of the code to backend servers (e.g. [10]) ensured that computationally expensive context recognition tasks could still be carried out in these devices. On the other hand, sensing pipeline optimization techniques (e.g. [81]) have facilitated on-the-device processing; this in-turn allows low latency real-time activity recognition.

However, there are some major challenges pertaining to the problem of multi-device activity recognition. In the next section I describe some of these challenges in detail

and how this dissertation tackles and addresses a well-identified subset of these challenges.

1.2 Challenges

In order to realise these automated ADL recognition systems using personalised devices and infrastructure sensing techniques, numerous challenges have to be addressed. In this section I list some of these challenges. However, before noting down the challenges, let us consider the following scenario which will assist in better appreciation of the challenges : *Monica, a social scientist, has recruited Joey for a study involving the understanding of ADLs performed by an individual through the day. One of the monitored ADLs is shopping. Monica has installed an ADL monitoring application on Joey's personal devices. On a particular day during the study, Joey walks into a shop wearing his smartwatch and carrying his smartphone. On determining Joey's location as 'in-shop', the ADL monitoring application in each of these devices identifies different aspects of Joey's shopping behavior – e.g. the smartwatch identifies picking gesture, while the smartphone determines Joey's behavior based on his trajectory. As Joey walks around the shop, inspecting and selecting items, Monica and Ross (Monica's colleague working on the same study) shadow Joey and note down all the items that he picks in the store. The shadowed data will be utilised to establish the ground-truth.*

Some of the challenges are:

- **Energy Consumption**: Smartphones, smartwatches and other wearable devices perform various tasks, one amongst which is activity recognition. Since activity recognition and lifestyle analytics might not be the only primary task of these devices (even though some devices have dedicated activity recognition modules), it is important to minimize the energy consumption of these devices, while performing the lifestyle analytics. For example, in the scenario mentioned previously, it is highly unlikely that Joey will be interested in

the study if the application drains out the battery in any of his devices within a few hours.

A major cause of battery drain during ADL tracking is the sensing process itself; every active sensor on a device drains the battery. The amount of drain is dependent on the sensor (e.g., the GPS sensor usually drains the battery faster than the accelerometer[13]), the sampling rate, the duty cycle etc. Extensive work has been done in identifying and implementing techniques which can reduce the energy consumption. Some of these techniques includes duty cycling [54, 81, 151], offloading [29, 77, 116], and inference [28, 89, 97]. For an application which has to continuously monitor daily life activities, certain sensors in the monitoring devices has to be turned *on*. Since sensing is expensive, the application has to ensure that the sensing process itself does not drain off the battery. A combination of existing techniques has to be custom-engineered to ensure the possibility of continuous activity monitoring. A constraint in a multi device environment is that different devices have different battery capacities and the sensor's battery drain might be different in two devices. This has to be kept in mind when choosing the energy conservation strategy. Since the smartwatch is the central component in the systems described in this dissertation, minimising the energy consumption is especially important because: (a) the smartwatch has a smaller battery capacity as compared to a smartphone and (b) some of the systems described in this dissertation rely on the smartwatch to identify gestures, which in turn identifies activities. Since activities might last for potentially long period of time, it is essential that the smartwatch's battery does not drain out before the end of the activity.

- **Accuracy**: Accuracy in identifying an ADL specific activity is an important factor for an end user. An application which consumes very little energy and provides unreliable prediction will provide no insight regarding the user's

lifestyle. In the example scenario, if the application running on Joey's smartwatch cannot identify any *item picks*, then the shopping activity monitoring system will not be of any use to either Monica or Joey. Alternately, if every sensor in the smartwatch is turned on, then the accuracy might be high, but the battery drain might be even more severe. Thus, based on application requirement, the balance between accuracy and energy must be maintained. Work such as [26] demonstrates how applications can balance between accuracy and energy consumption. In this dissertation, we describe two systems, one that captures images of the individual's food plate during a meal, while the other identifies all the items picked by a customer in a retail store. Both these systems have different accuracy needs. In case of capturing images during the eating activity, we need to capture just one useful image of the food item, across multiple gestures. In contrast, in shopping, the goal is to capture the location of every individual pick gesture. These requirements permit or disallow certain types of energy-saving optimizations (e.g., turning off the expensive gyroscope sensor may be possible when the activity involves multiple repeated gestures).

Accuracy errors might creep in if the classification model is erroneous or if the choice of sensors to recognise the activity is incorrect. For example, understanding eating gestures might be possible using accelerometer data of smartwatches, while understanding locomotive states might be possible using accelerometer data of a smartphone. However choosing a smartphone for eating recognition or a smartwatch for locomotion recognition might cause significant accuracy drop. Accuracy will also be compromised if there is error in correlating between the devices. Work such as [85] and [127] shows how multiple devices can be used to collaborate across devices and thus have high accuracy.

- **Near Real-time Processing**: An application requirement for ADL monito-

ring might be to perform near-real time sensing and processing. The sensing and processing might occur either on the device itself or the processing might be partly offloaded to another device. To understand the importance of near-real time processing, in the scenario described previously, if Joey should be able to receive store level promotions, where a promotion is determined based on the items that Joey picks, then the processing of *pick-identification* should occur in near-real time (within a few seconds of the occurrence of the pick gesture). With continuous improvement in the hardware, processing sensor data on smartwatches and smartphones is possible. Work such as [16] and [81] demonstrates the possibility of continuous sensing and processing sensor data on smartphones, while this dissertation (Chapter 4) shows the possibility of real time activity recognition on the smartwatch.

For the two systems presented in this dissertation, near real time identification of the hand gesture is important as this allows us to trigger other sensing modalities. For the food journaling system, the real time hand gesture identification triggers another sensor – the camera on the same device, while for the shopper's item interaction monitoring system, the inertial sensor on one device triggers the sensors on the other device.

- **Diversity across users**: A major challenge in activity recognition is that a highly accurate system needs personalized models – training data from the monitored individual. Systems using personalized models do not address diversity exhibited by various individuals. For example, Joey's item picking style might be very different from Monica's. A model that has been trained on Monica's picking gesture might have low accuracy in identifying Joey's picks. The problem with building personalised models is that systems which have been built on personalized data do not scale easily. Work such as [63] shows that data from the community can be used to reduce personalization. An alternate approach for handling user diversity might be to use multi-modal

sensing, where the same context might be established through different sensing modalities and devices.

- **Data Annotation**: The process of collecting labeled ADL data is a challenging task as multiple possible sources of error exist in the labeling process. For situations where ground truth data has been video recorded and labels are extracted from the video (e.g. [145]), inaccuracy in synchronising the sensing device and the ground truth collection device will lead to incorrect segment annotations. Moreover, the process of data annotation from videos by itself is a tedious manual task, which requires extensive effort. Alternately, for scenarios where labeling is done by shadowing the participant (e.g. [130]), the additional problem of incorrectly labeling (e.g., for ground truth labeling, Monica marks Joey's pick once Joey has displaced the item from the shelf, while Ross marks that a pick gesture is taking place even before Joey has touched the item.) and missing out labels exists (e.g. since the shadower has to make split second decisions about labels that has to be marked, in case wrong label marking is done, then going back in time and changing the label is not possible.) Additionally, if multiple labels have to be marked at the same time, then some labels can get missed. A third approach is to utilise annotations provided by users themselves (e.g. [11]). However, issues related to false or under-reporting might arise when users self annotate the data [43].

  For the activities described in this dissertation, the key annotation that had to be performed in real time was capturing the hand gesture. Since the hand gesture can be noisy and individuals can be unpredictable (e.g., an individual raises his hand to consume a spoonful of food, but before putting the food in the mouth he gets engaged in a conversation) we had to ensure that incorrectly marked labels could be discarded, else the incorrectly marked gestures would affect the overall system's performance.

- **Unsupervised Learning**: Another issue with data annotation is that in case

of large datasets, it is almost impossible to manually annotate each and every data instance (e.g. identifying and labeling every item that any customer shopping in a giant supermarket picks). For such scenarios, some automated labeling techniques have to be identified which can label the data in an unsupervised manner. Works such as [64] and [65] have shown that in the field of image processing, a small corpus of training data can be used to identify items in a large corpus of unlabeled data. Techniques similar to this can be used in the field of activity recognition. Alternately, work such as [63] has shown the possibility of labeling data based on how a community performs, while [158] has shown that automatically learning models from the web can help in unsupervised clustering.

- **Privacy**: Data collection from sensors on personal devices naturally raises privacy concerns. Sometimes the privacy implications might not even be obvious (e.g. – sensor data collected from the smartwatch's inertial sensor might be capable of determining a person's personal details [102, 131, 148, 149]). Therefore, privacy aspect should always be considered while collecting sensor data from any personal or infrastructure sensing devices. While it might not be possible to tackle every privacy threat, for many scenarios, basic prevention techniques such as deleting the raw sensor trace or introducing random noise in the trace might be useful. Alternately, instead of providing fine grained context, a high level context might be provided. [157].

In this dissertation, I have primarily addressed a subset of the above mentioned challenges – specifically, the challenges related to *energy consumption*, *accuracy*, *near real-time processing* and *diversity across users*.

1.3 Thesis Statement

Now that I have discussed about the opportunities and challenges pertaining to identifying activities of daily living using wearable, mobile and IoT sensors, in this dissertation I show that:

> It is indeed possible to harness the multi-modal sensing capabilities of commercial, off-the-shelf mobile, wearable, and IoT devices to derive accurate, and fine-grained insights about multiple aspects of an individual's daily lifestyle activities and behavior (with eating and shopping used as exemplars) in near real-time.

Building and easily deploying such systems using off-the-shelf devices involves identifying appropriate sensing modalities and optimising the usage of the appropriate sensor in each of the devices. Since the corpus of possible devices, sensors and target audience is large, there are many research questions that need to be answered. This dissertation establishes the thesis via the following steps:

1. First, it determines the characteristics of regular daily life activities with shopping and eating activities as examples. Since every activity has its own level of complexity (e.g., shopping may be complex, as it involves both gestural and locomotive actions, while eating may be complex because the mode of eating and the content being consumed cause the eating behavior to vary significantly), understanding these characteristics help identify the design goals that a lifestyle monitoring system should achieve.

2. Next, for each of the monitoring systems, it determines whether fine-grained context of the daily life activity can be determined by (a) one or more sensors in a single device or (b) fusion of sensor data from multiple personal devices or (c) fusion of sensor data from personal and infrastructure devices.

3. It then presents two solutions (i) *Annapurna*: a system for automated food journaling, and (ii) I^4S[1] a system for identifying in-store item interaction.The two systems are designed to infer the fine grained details of each of the daily life activities either through analysis of data from multiple sensors on a single device, or via analysis of sensor data from multiple devices.

   Annapurna demonstrates the possibility of utilising multiple sensors in a wrist worn device (smartwatch) for fine-grained context inference. More specifically, data from the inertial sensor on a smartwatch acts as a gestural marker in *Annapurna* to trigger the smartwatch's camera and capture images of the food being consumed.

   I^4S demonstrates that in certain scenarios, a single device might not be capable of identifying fine-grained context of a daily life activity. In such a scenario, finer contextual insights can be obtained by fusing data from several sensors embedded in multiple devices.

4. Additionally, this dissertation explores the possibility of person-independent identification of cognitive/mental context associated with the daily life activities. Through exploration of data from personal devices, this dissertation demonstrates an approach to address user-diversity while determining user's cognitive state during the daily life activity.

There were some interesting findings during the process of establishing the thesis.

First, it is well known that energy can be conserved when a cheaper sensor adaptively turns on a more energy consuming sensor [151]. For *Annapurna* – the automated food journaling system, we found that continuous video capturing drained out the watch in approximately 80 minutes. We thus used the inertial sensor as a gestural marker to turn on the camera. However, we found that turning on the camera after a gesture was detected resulted in capturing of images where the object of interest was not visible . This was because of the latency which existed

[1] pronounced i-foresee

in the entire image capturing process. The delay in turning on the camera after detecting the gesture results in frames where the object of interest is missing. We thus had to look for alternate image capturing approaches which could capture the object of interest accurately (with some possible loss in energy).

Second, in I^4S, even though both the smartwatch and smartphone were capable of BLE scanning, I had to eventually utilize the smartphone's BLE scan information to determine the shopper's in-store location. This decision was driven by my finding that (i) beacon miss rates were typically higher in the smartwatch and (ii) the hand movement during shopping gesture had very short duration, which resulted in noisy location prediction.

Third, standard image recognition techniques, with decent performance in identifying regular items might not perform well in identifying partially hidden objects which have been captured from unorthodox angles and with motion blur. In *Annapurna*, images of the food plate was captured using a camera mounted on the smartwatch. We found that state-of-the-art image recognition algorithms were not very effective in identifying such images. We thus switched over from identifying the item to a heuristic based determination that the object of interest (food) might be present in the image.

1.4 Research Contribution

This dissertation explores the use of commercially available, off-the-shelf devices to infer and analyse Activities of Daily Living (ADLs). It uses eating and shopping as two example ADLs (eating and shopping fall in two different ADL classes) and describes techniques to identifying fine-grained details of the two ADLs. The main contributions of this dissertation can be summarized as follows.

- **[Contribution 1] ADL monitoring and Analysis**: This dissertation develops novel techniques to identify additional, fine-grained attributes of daily life activities. It utilises inertial sensor data from the smartwatch (an off-the-shelf, fixed positioned, body-worn device) to identify natural gestures (hand-to-mouth in case of eating, or reaching-for-item in case of shopping). The activity specific natural gesture acts as a marker for turning on additional sensors (either on the smartwatch or other devices) to capture finer details of the activity. This has been demonstrated by: (i) building *Annapurna*, a food journaling application, which detects the "eating" activity and automatically creates a food journal and (ii) designing I^4S, a system to detect items that a shopper interacts with while shopping. This dissertation shows how this concept of gesture-based sensor triggering can be implemented in different ways. For eating, where the gesture itself is repetitive, I have demonstrated how energy can be saved by triggering the gyroscope sensor only in the finer-layer of a two-tier gesture recognizer, and also how useful food images can be obtained by triggering image capture within individual gestures (without needing to continue such image capture across gestures).

  While developing these applications and designing the techniques, I have shown how real world system challenges can be addressed. For example, for the food journaling application, I have shown techniques applied to reduce energy consumption, whereas in the item interaction system, I have shown how the choice of sensing modality affects accuracy.

- **[Contribution 2] Fine-grained Monitoring using data from multiple sensors in a single device**: Work reported in [104] and [145] has demonstrated the possibility of utilising a single class of sensor (inertial) in a smartwatch to identify a natural gesture, which in-turn indicates a specific activity period (smoking sessions or eating periods). This dissertation demonstrates that finer-details of a specific activity can be determined if the natural gesture

can be utilised to trigger the data collection from additional sensors classes in the device. Specifically, for the hand-to-mouth gesture during eating, this dissertation demonstrates the possibility of identifying finer details about the eating activity and building an automated food journal if the natural gesture can trigger additional sensor – the smartwatch's camera.

Existing single-device solutions for eating gesture recognition and food journaling either only identify the eating gesture [145] or capture images of food being consumed [76, 125]. To the best of my knowledge, there is no automated food journaling system built using a personal off-the-shelf device, which not only identifies the eating gesture, but also opportunistically captures images of the food being consumed and builds a food journal using the captured images. This dissertation has created *Annapurna*, a system which not only identifies the eating gesture, but also captures the images of the food consumed. *Annapurna* is built using an off-the-shelf commercial smartwatch (Samsung Gear 1), which has a built in camera. The inertial sensor of the smartwatch determines the eating gesture and episode, while the smartwatch's camera opportunistically captures the images of the food consumed.

Various system level contributions emerged during the building of *Annapurna*: - (i) The hand-to-mouth gesture during an eating episode is a periodically repetitive action. *Annapurna* shows that using a two tier classification approach, where a lower energy-cost classifier on identifying the possibility of hand-to-mouth gestures triggers a more expensive classifier, can reduce the energy consumption of the system. (ii) *Annapurna* shows that innovative design choices made during the energy intensive [74] image capturing process can reduce the energy consumption of the process.(iii) Given that images captured during eating may be irrelevant or unusable (e.g., food item absent, image too blurry), *Annapurna* shows how simple image processing techniques (e.g., edge detection, depth estimation) can be effective in selecting the relevant images and significantly reduce the volume of data that needs to be

transferred. Chapter 4 provides system details of *Annapurna* along with the rationale behind the design choices taken in the system implementation.

- **[Contribution 3] Fine-grained monitoring using multi-modal sensing across multiple devices**: For certain scenarios, either a fixed-positioned smartwatch might not be capable of identifying the complete, fine-grained context (e.g. to identify if a person is standing, sensors attached to an individual's leg or smartphone in the individual's trouser pocket might provide higher identification accuracy as compared to a wrist-worn smartwatch) or the sensors in a device might not be robust. In such scenarios, sensor data from additional personal devices might be useful in determining fine grained context. This dissertation demonstrates that finer-details of a daily life activity (shopping) can be determined if the natural gesture identified by a smartwatch can be utilised to trigger data collection from other personal devices. Additionally, this dissertation also demonstrates that instrumented environments can further assist in determining finer details of a context.

  Through the shopping activity monitoring as an example, this dissertation shows how multi-modal sensing can assist in fine-grained shopping context identification. For a physical store owner to obtain fine grained information about shopper's interactions with items (e.g., 'pick', 'evaluate', 'return to shelf' or 'put in cart'), we designed I^4S . I^4S, a low-cost system, not only uses the inertial sensors on the smartwatch to identify the "pick" gesture (a natural shopping gesture) of the shopper, but also utilises infrastructure sensors (BLE beacons) to identify the location from where an item is being picked.

  To achieve this fine-grained shopping interaction tracking, I^4S fuses sensor data from multiple devices – picking gesture is determined by the inertial sensors of a smartwatch. The pick gesture triggers the inertial sensor on the smartphone (to determine locomotion state) and the BLE scan information (to determine store location). In I^4S, different sensors from multiple devices are

combined to deduce different context. For example, the inertial sensors of a smartphone and smartwatch are jointly used to determine shelf-level location, whereas the inertial sensors and bluetooth sensors from the smartphone are combined with inertial sensors on a smartwatch to determine relevant ”pick” gestures (without suffering from high false positives). Even though the idea of location instrumentation has been studied in smart homes (where every object is tagged with sensors) and shops (with dense active-RFID deployments), these deployments can be expensive. I^4S demonstrates that it is possible to determine deeper individual-specific context in a modestly instrumented store with lower deployment cost as compared to existing instrumentation approaches. Chapter 5 provides the details of I^4S along with various rationale for several design choices.

- **[Contribution 4] Behavior determination**: Once the physical ADL monitoring techniques are in place, various behavioural insights about an individual can be extracted. For example, a retail owner might be interested in identifying customers requiring assistance. To realise this, we explored approaches to determine the cognitive state and behavioral context exhibited by a user during a daily life activity – shopping. Since individuals can exhibit diverse physical behavior while having the same underlying cognitive state or behavioral intent, this dissertation explores the possibility of accommodating the behavioral diversity exhibited across individuals. In contrast to past work that uses demographic attributes explicitly to capture such diversity, I've proposed *CROSDAC*, an implicit data-driven approach to establish the number of distinct 'styles' by which such diversity is manifested. *CROSDAC* identifies the behavior of an individual by comparing the behavior pattern with other “similar” individuals. Chapter 6 provides the details of *CROSDAC* and how it determines the shopper's behavioral state.

1.5 Project Contributions

For all my studies, I have collaborated with multiple colleagues and faculties, who have helped in stages right from idea formulation to user studies to final ADL determination. In this section, I list down contributions of various colleagues for the various projects. (Note: Archan has been involved in the planning, refining and improvement of the techniques in all the projects mentioned below, while Rajesh and Youngki have been involved in *Annapurna* and I^4S.)

Annapurna - For the development of *Annapurna*, Vigneshwaran Subbaraju (A*Star) has contributed significantly in various stages of the project - he was involved in various stages of the system design planning, developing image capturing through preview mode and model improvement and also data collection. For the project, almost everyone in the lab has contributed by providing smartwatch's sensor data for techniques which have worked or failed. Table 1.1 captures the module/activity level effort by various non-faculty contributors:

	Sougata	Vignesh
System Designing	60%	40%
Activity Recognition	75%	25%
Image Capturing	50%	50%
Image Processing	60%	40%
Data Collection	50%	50%
Data Processing	60%	40%

Table 1.1: Project Contributions – *Annapurna*

	Sougata	Karan	Meera	Vignesh
System Designing	50%	20%	15%	15%
Pick Detector	80%	20%	-	-
BLE localizer	100%	-	-	-
In-shelf localizer	25%	75%	-	-
Data Collection	33%	33%	33%	-
Data Processing	80%	20%	-	-

Table 1.2: Project Contributions – I^4S

I^4S - For I^4S, the multiple planning phases involved thought contributions from Vigneshwaran Subbaraju, Meera Radhakrishnan (SMU) and Karan Grover (IIITD). Meera and Karan have been actively involved in various phases of data collection.

	Sougata	Vignesh	Dipanjan	Dipyaman
Approach Planning	40%	-	30%	30%
Data Collector	100%	-	-	-
Data Collection	100%	-	-	-
Approach Implementation & Testing	75%	25%	-	-

Table 1.3: Project Contribution – *CROSDAC*

Karan has also been involved in using quaternion data from smartwatch sensor to determine position within rack. Table 1.2 captured the module/activity level effort by various non-faculty contributors for I^4S.

CROSDAC - For *CROSDAC*, I received ample mentoring from Dipanjan Chakroborty (IBM IRL) and Dipyaman Banerjee (IBM IRL). Vigneshwaran has again been involved in the planning and testing of various possible approaches. Table 1.3 captured the detailed effort by various non-faculty contributors for *CROSDAC*.

1.6 Dissertation Roadmap

In my exploration of the open challenges described in the previous subsections, I have taken an experimental approach, where I have built prototypes, conducted user studies and performed experiments to validate my ideas. Given the relatively small user sample sizes used in my studies, I have additionally performed sensitivity studies to demonstrate that my results and insights are robust, and likely to apply to a wider population. All these details are reported in this dissertation. The rest of the dissertation is organised as follows:

Chapter 2 describes literature which is most relevant for this thesis. The chapter is divided into four parts, where I first describe the existing activity recognition works. Then I specifically look into literature pertaining to eating and shopping activity recognition using wearable, mobile and infrastructure sensors. I finally discuss about literature related to mobile and wearable based behavior recognition.

Chapter 3 presents a brief background of ADL monitoring with examples of possible motivating scenarios. I explain the scenarios and show how systems that

we have developed can address the scenarios. I provide a high level overview of the systems and techniques that we have exploited while providing solutions to the explained scenarios.

In Chapter 4, I describe the automated food journaling application - *Annapurna*, which captures eating and diet details, and shares such captured details with an individual consumer via a personalized Web portal. In the chapter, I first describe the need for having a food journaling application and then provide the system overview. Since eating is a complicated activity, I specify the design goals of *Annapurna*, along with the design of the system. *Annapurna* was built over multiple iterations; where a user-study was performed and lessons learnt from the user study was used to improve the design in the subsequent iteration. In the chapter I describe the user-study details for every iteration and explain the lessons learnt and how it helped in improving the system. Finally, I describe the web application that was presented to the end-user, where the user could track food items consumed.

Chapter 5 presents I^4S – an approach that I have explored to identify fine grained in-store item interactions. I^4S identifies (i) the "picking" gesture - a *fine-grained* shopping specific gesture exhibited by shoppers and (ii) location from where the item was picked. I^4S utilises a smartwatch, a smartphone and infrastructure deployed BLE beacons to identify the location from where an item was picked. In the chapter, I first explain the system overview, the design choices taken, and the overall approach in building the system. An in-store user study was performed to determine the feasibility of identifying the in-store item interactions. I explain details of the user study along with the performance results of various components of I^4S.

In Chapter 6, I present a crowd-scale, non-person specific, recognition framework, called *CROSDAC*, to tackle the problem of inferring the intentions and attitudes of individual in-store consumers, using mobile sensing data. In the chapter, I describe two user studies that were conducted to validate *CROSDAC*. This is followed by the analysis of the data collected in determining shopper's intentions. Finally, I end the chapter with some open problems and challenges.

In Chapters 7, I describe the various possible extensions of the current systems as well as other viable future research directions and in Chapter 8, I recap the systems and provide my closing comments.

Chapter 2

Literature Review

Activity Recognition has always been of interest to researchers. With the high availability of personal mobile and wearable devices as well as infrastructural sensors, activity monitoring techniques have transmuted from manual monitoring approaches to automated and to almost unobtrusive techniques. Since activity recognition lies at the foundation of all my contributions, in this chapter, I will first discuss about some activity recognition techniques that have been developed, followed by a high level overview of some existing ADL monitoring systems. The bulk of my work is largely related to two ADLs - eating and shopping. Subsequently, I will present works which either focus on alternative ways for monitoring these two specific daily life activities or which use sensors/features in ways very similar to our monitoring approaches, but to monitor other types of ADLs. Finally, I will discuss works which have attempted to use data and information from sensing devices to determine intent or behavior of end users. The rest of the chapter is arranged as follows:

Section 2.1 discusses multiple activity recognition techniques developed and tested by researchers and how these solutions address system related challenges. This section will also present how this activity recognition is used for general ADL monitoring.

Sections 2.2 and 2.3 continues discussing about various ADL monitoring techni-

ques developed by researchers. Section 2.2 discusses about eating monitoring techniques. It also provides details of the literature existing in the area of image recognition as our eating ADL monitoring technique relies on image processing and identification. Similarly, Section 2.3 primarily discusses about literature related to various existing shopping ADL monitoring techniques, it also discusses literature related to indoor localization as well as techniques to understand human behavior using mobile and wearable devices that have been explored in various studies.

Finally, Section 2.4 shifts from literature related to identifying the physical ADL monitoring activity to literature related to identifying an individual's cognitive state and behavior through analysis of sensor data.

2.1 Activity Recognition

Sensor based activity recognition has been researched for several years, with activity recognition using sensor data from smartphones and wearable devices being one of the newer trends in this research domain. The most commonly used sensor in the smartphone for activity recognition is the *accelerometer*. Accelerometer based activity recognition techniques were tried and tested even before accelerometers made its way into smartphones. Before the smartphone era, accelerometers were attached to an individual and activity recognition was performed by processing the data from these sensors [11, 123]. However, with technological advancements leading to the introduction of accelerometers in smartphones, works such as [81, 86] demonstrated the use of smartphone for simple activity recognition. Since reducing the energy consumption during activity recognition is a challenge (Section 2.1.1 details several approaches to address this challenge) and the energy consumption of the accelerometer is lower than several other smartphone sensors [13, 112], it is advantageous to use the accelerometer for smartphone/wearable based activity recognition. However, since accelerometer might not be the optimum sensor for activity recognition in certain situations, other sensors have also been utilised to monitor activities – e.g.,

the microphone to monitor (a) general activities [140], and (b) bathroom related activities [23], or the GPS sensor for simple activity recognition [72]. Section 2.1.2 describes several sensing modality utilised to monitor numerous daily life activity monitoring categories.

As an alternative to utilising sensors from personal devices, activity recognition through infrastructure sensors is also possible. Multiple activity monitoring approaches using infrastructure sensors have been proposed by researchers. The use of video feeds to determine individual-specific activities were explored in [20] and [156], while the possibility of recognizing such activities via passive monitoring of RF signals have been addressed in [1, 153]. The major advantages of these techniques are that they are device-free techniques and do not require any on-body devices for the activity recognition. However, since the device performing the activity recognition is not a personal device, the activity prediction might also lead to privacy leaks. Additionally, they are often designed for specially instrumented environments.

2.1.1 Addressing Challenges in Activity Recognition

Energy Consumption: Since rapid battery drain in a smartphone affects usability, reducing the energy consumption of a smartphone has always been of interest to researchers. Several techniques have been proposed to ensure that the activity recognition (even accelerometer based) does not cause a noticeable battery drain (or battery drain in any other personal device) [28, 29, 54, 68, 77, 81, 89, 97, 116]. In scenarios where continuous context monitoring is required, works such as [105] and [151] demonstrate that using a cheaper sensors to turn on a more expensive sensor saved energy. In these studies, the cheaper sensor identified a certain context, which acted as the marker to turn on the more expensive sensor to monitor additional context. We have used a similar technique in *Annapurna*, where we turned on the camera (the more energy hungry sensor) to capture images of the food consu-

med only when we received contextual markers – identification of the eating gesture by the accelerometer sensor.

Accuracy: In addition to energy, activity recognition systems must ensure that the activity predicted by the system is accurate. It has been well established that personalised models for activity recognition perform better than general models [11]. However, collecting personal data to create personal models is labor intensive. Other than personalised data, for the same participant, the position and orientation of the device affects the accuracy [70] – if training data is collected in a certain device orientation and the device's orientation is different during testing, then the accuracy of the system will be affected. Other factors which affect the system's accuracy includes: number of sensors used in the data collection/ activity prediction or the choice of classifier [159]. Sometimes, a single sensor (or device) might not be robust enough to determine an activity. Work such as [69] and [85] use sensors on multiple phones to improve on the context recognition – accuracy of speaker identification, while work such as [116] utilizes infrastructure sensors to improve on the accuracy along with energy conservation. The approaches adopted by researchers which have been highlighted here are a few possible approaches for the personal devices to either save energy or improve accuracy or both. Extensive literature exists, showing multiple possibilities for improving system's performance. Our shopping behavior identification approach, *CROSDAC*, demonstrates the possibility of achieving reasonable accuracy in identifying behavior, without any personal data, while I^4S, demonstrates the possibility of improving identification accuracy by using sensor data from multiple devices.

2.1.2 Identifying and Monitoring Specific Daily Life Activities

I next discuss some ADL monitoring systems and techniques developed by researchers, which utilises sensor data for various sensing devices (Detailed summary provided in Table 2.1). Sensing devices utilised for ADL monitoring range from

Study	Study Type	Monitored Activity	# Sensing Devices	Device Type	Device Details	Device Position	Sensor(s)	Obtrusive	Processing
[3]	In-Lab	Eating	4	Custom - personal	Sensor attached to body	Wrist x2 Arm x2	Accelerometer	Yes	Offline
[31]	In-Lab	Eating	1	Custom - personal	Smartwatch like device	Wrist	Gyroscope	Yes	Offline
[104]	In-the-Wild	Smoking	1	Custom - personal	Smartwatch like device	Wrist	Rotation vector	No	Real-time
[46]	In-the-Wild	Toothbrushing	2	Custom - personal	Smartwatch Custom Toothbrush	Wrist	Accelerometer, Gyroscope, Gravity, Magnetic, Acoustic	No	Real-time
[78]	In-the-Wild	Driving	2	Smartwatch - personal Smartphone - personal	Smartwatch Smartphone	Wrist Car dashboard	Accelerometer, Gyroscope, Magnetometer	No	Near real-time
[87]	In-the-Wild	Sleep	1	Smartphone - personal	Smartphone	in Bedroom	Accelerometer, Microphone, Ambient light sensor, Screen proximity sensor, Running process, Battery state, Display screen state	No	Offline
[150]	In-the-Wild	Activity Conversation Sleep	1	Smartphone - personal	Smartphone	NA	Accelerometer, Proximity, Audio, Light, Location, Co-location, Application usage	No	Offline
[143]	In-the-Wild	Several in-home-activities	161	Custom - infrastructure	State Change sensors attached to devices	Several items in house	State change sensor	No	Offline
[124]	Controlled In-the-Wild	Breathing	2	Off-the-shelf- infrastructure	2.4 GHz Transmitter and Receiver	Mounted on wall	NA	No	Real-time
[118]	Controlled	Shopping	2	Smartwatch - personal Smartphone - personal	Smartphone Smartwatch	Pant pocket Wrist	Accelerometer, Gyroscope, Magnetometer, Step counter, Battery temperature, Light sensor, Audio, Heart rate	No	Offline

Table 2.1: Comparison of Various Dailly Life Activity Monitoring Systems

personal devices to infrastructure devices. The type of personal devices also vary – while some studies used off-the-shelf devices like smartwatches and smartphones, for others, the authors created custom hardware. I first discuss studies which utilises a smartwatch or smartwatch like device: the use of wrist worn sensors/devices to determine the eating activity was demonstrated in [3], and [31]. The authors in [3] did not develop a custom device, but rather attached the sensors to the upper limbs to monitor the eating activity. Besides eating, the feasibility of monitoring smoking activity through a custom smartwatch was demonstrated in [104], while sensor data from an off the shelf smartwatch and smartphone has been utilised to determine the driving behavior (e.g., [78]) and shopping activity (e.g., [118]). For all of these studies, the inertial sensor is primarily used as these studies require monitoring the hand's motion. Similar to the previous studies, a smartwatch based approach to monitor toothbrushing activity was explored in [46]. In addition to the smartwatch, the authors developed a custom toothbrush with a magnet attached. The magnet allowed the researchers to monitor the hand orientation of an individual while the individual used the toothbrush. All these researchers have utilized the smartwatch (or almost equivalent device – not necessarily off-the-shelf) for the activity recognition. Other than smartwatches, smartphones have also been used for various lifestyle analytics. Work such as [87, 25] have used the accelerometer, the microphone, and the light sensor on the smartphone to determine sleep duration and quality of sleep, while smartphone usage pattern to determine sleeping period and wake up period was explored in [150]. Besides sleep, the authors in [150] also utilised the smartphone to identify other activities and individual's conversations.

ADL monitoring using infrastructure sensors has existed even before personal devices were used for ADL monitoring. A lot of effort has been made towards smart homes for ADL monitoring. The authors in [143] demonstrated that attaching simple sensors to devices in a house can be used to monitor daily life activities in an instrumented home. These sensors could identify when a certain device was used by the occupant in the house. Alternately, the possibility of identifying activities

inside a house using RF signal has been explored in [1]. However, the problem with these only infrastructure based techniques is that it is not possible to associate an ADL with a particular user in the case of multi-occupant scenarios.

2.2 Eating Activity Recognition

I next focus specifically on eating detection. In addition to eating detection, since my work involves the use of image capturing, I also provide some details of existing image recognition techniques.

Eating Identification and Monitoring: Online food journals such as [95] allow users to manually note down all food items consumed, while applications such as [146] allow self reporting of food consumed through a smartphone application. However, literature shows that self reporting leads to under reporting (e.g., [43, 110]). To overcome this, automated food consumption identification/ monitoring, and journaling approaches have been proposed by various researchers, where techniques vary from using instrumented locations [22], tabletops [166] etc. to utilising one or more mobile and wearable devices. These devices can either be off the shelf [133] or custom made [31] and can use techniques such as inertial sensors [145] or microphones [121] or image / video [125] or even a fusion of multiple techniques [76]. Additionally, there has been work to identify the items consumed [66]. Since my work revolves around using mobile and wearable devices, in this section, I introduce relevant eating detection and monitoring systems which primarily utilise mobile and wearable devices.

A popular food intake monitoring approach is through the use of wearable sensors with acoustic monitoring capabilities. Work involving acoustic sensing either detects chewing or swallowing or both. Identification of the chewing sound, an indicator of food consumption has been explored in [4] and [160]. Additionally, the possibility of identifying the texture of the food from the chewing sound was demonstrated in [160]. Several chewing detection studies utilise an ear worn device

Study	Study Type	Device Used	Number of Devices	Sensors Used	Sensor Location	Obtrusive	Automatic	Approach	Study Details	Eating Identification	Food Recognition	Food Journaling
Annapurna	controlled in-the-wild	off-the-shelf	1	S1, S2, S4	wrist	No	Yes	hand movement triggered image capturing	controlled - 21 In-the-wild - 7	Yes	Partial	Yes
[3]	in-lab	custom	2	S1, S2	wrist, arm	Yes	Yes	hand movement capturing	2 subjects 384 gestures	Yes	No	No
[5]	in-lab	custom	3	Arm - S1,S2,S3 Ear - S5 Neck - EMG	arm, ear, neck	Yes	Yes	hand movement & food consumption sound	4 subjects 1020 gestures	Yes	Partial	No
[31]	in-lab	custom	1	S1,S2,S3	wrist	No	Yes	hand movement capturing	98 subjects 188 meals	Yes	No	No
[76]	restaurant	custom	1	S4,S5	ear	Yes	Yes	chew detection triggered image capturing	6 users 1 meal	Yes	No	Possible
[88]	in-lab	off-the-shelf	4	Ear - S4 Wrist - S1,S2,S3 Glass - S1,S2,S3	ear, head, wrist	Yes	Yes	continuous capturing	6 users 72 hours 40 food items	Yes	Yes	Possible
[95]	NA	NA	0	NA	NA	No	No	self reporting	NA	No	No	Yes
[121]	in-lab	custom	1	S5	neck	Yes	Yes	food consumption sound	14 subjects 15 minutes	Yes	No	No
[125]	in-the-wild	off-the-shelf	1	S4	neck	Yes	Yes	duty cycled image capturing	6 users 2 weeks	No	No	Yes
[145]	in-lab in-the-wild	off-the-shelf	1	S1	wrist	No	Yes	hand movement capturing	in-lab - 20 real-world 7 and 1	Yes	No	No

S1 - accelerometer, S2 - gyroscope, S3 - magnetometer, S4 - camera, S5 - microphone

Table 2.2: Comparison of Various Eating Activity Recognition Techniques

to identify chewing. Alternately, researchers have developed custom devices to be worn on the neck/throat with the goal of detecting swallowing. The possibility of identifying the swallowing activity has been explored in [128] and [121]. Both of these approaches utilized a neck-worn device. More recently, the use of neck worn wearable sensors not just for chewing/swallowing detection, but also identifying the type of food has been studied in [18]. Most of the acoustic food monitoring systems utilise custom hardware which has to be attached to the body and have been tested in lab conditions. The devices are usually obtrusive. It can be argued that future earphones can be attached with the hardware. However, the user has to wear the device during food consumption, which might not be acceptable, especially in social settings, where multiple people are dining together. Additionally, little work has been done to understand the effect of real-world noise on these systems.

An alternate approach for food intake identification and monitoring is by utilising visual information – images or videos. The feasibility of image capturing through a smartphone's camera was demonstrated in [125]. The work relied on using images captured by a smartphone's camera while the phone suspended across the user's neck using a lanyard. Even though the system was automatic, it was obtrusive. Work such as [168] have removed the obtrusiveness by asking the users to manually capture the image of the food plate at the start and end of a meal. The authors not only identified the food items consumed, but also the quantity of consumption. Similarly, recognising the food consumed by the user from the images obtained from the camera of a smartphone has been studied in [52] and [66]. However, similar to [168], both these techniques require the user to explicitly acquire or label the images of the food and then they identify the food item.

Food intake identification techniques which are closest to my work are the ones that utilise the inertial sensor data from wearables to identify eating gesture. The inertial sensor based food identification approaches utilise either the accelerometer or the gyroscope or both. Early, non-smart wearable based eating gesture detection was demonstrated in [3]. In this work, the authors attached four accelerometer sensors

in several arm positions to identify the eating gesture. Even though the system was obtrusive, the authors demonstrated the possibility of utilising the accelerometer for accurate eating detection. More recently, the possibility of utilising accelerometer data from two off the shelf devices – a smartwatch and a smartglass to determine eating gesture in a controlled study was demonstrated in [161]. With an accuracy of 89%, the authors demonstrated the possibility of identifying eating gesture using two off-the-shelf devices. The accelerometer data from a single wrist worn device could also detect eating activity in real world settings was shown in [145]. The study was performed in both controlled and real-world setting. The performance of the system was slightly lower than the system utilising two wearable devices. All the above mentioned systems utilise only the accelerometer for eating gesture identification. Alternately, the possibility of utilising the gyroscope for identifying eating gesture was explored in [31]. The gyroscope identifies the rotation of the wrist during the eating activity to identify the eating gesture. The work relies on custom hardware for hand rotation detection. The fusion of both the accelerometer and gyroscope data for eating gesture detection is explored in [32]. Similar to [32], our work (described in Chapter 4) utilises the data from both the accelerometer and gyroscope to determine the eating gesture. However, we utilise an off-the-shelf device for the gesture identification. Additionally, we also capture the image of the food consumed when an eating session is identified.

Till now I have discussed some techniques which use a single class of sensor. I next discuss techniques which uses multiple sensor classes. The multiple sensor classes either reside on the same device or reside on multiple devices. The possibility of identifying eating gesture using a microphone attached to a custom ear-worn hardware was demonstrated in [76]. Once eating has been detected, the device turns on the camera embedded in the earpiece to capture images of the food consumed. The device has been tested in a real world setting (university restaurant). However, the authors did not show the performance of the system while the individual performed other similar activities. In terms of approach, this work closely resembles our

work, where we use inertial sensor data instead of the microphone data as a trigger to start capturing images through an embedded camera in the same device. However, our study did not rely on a custom device. Other than multiple single-device sensor based food identification, the possibility of utilising multiple sensor classes on multiple devices to identify the food consumed was demonstrated in [88]. The authors used 2 wrist worn, a head worn and an ear worn devices for the study. In the controlled study, the authors demonstrated the possibility of identifying multiple food types using multi-sensor data fusion. Table 2.2 shows a detailed comparison of several approaches discussed in this section.

Since *Annapurna* captures images and identifies the *best* images amongst the captured images, I next discuss some possible image recognition techniques which has been implemented.

Image Recognition: Automatic object recognition on mobile phones has been reported in work such as [91], where a smartphone camera is used to identify medication packages. The system first extracts robust features from the images and then uses these features for object detection. In computer vision, the Scale-Invariant Feature Transform (SIFT) [79] and Speeded-Up Robust Features (SURF) [12] are some common methods used to identify robust features from images of objects. Machine learning classifiers (trained using a large corpus of images) are then used to recognize the objects from the extracted features and some addition image descriptors. The classifiers are trained using a large corpus of images. Deep learning [14, 30] using convolutional neural networks [57, 27] is commonly used for object recognition from images. However, most deep learning frameworks are designed for servers. Work such as [60] provides a measurement study of the resource requirements and constraints involved in implementing deep learning on mobile and wearable platforms. Alternate approaches such as [24] (which performs continuous recognition and tracking of traffic signs) offload the image recognition tasks to the server.

2.3 Shopping Activity Recognition

Monitoring the shopping activity has been of interest to the marketing community, as it provides various insights about the activity itself. An approach to identify the shopping interactions is through observations [71, 154]. However, manual observation approaches are labor intensive. To overcome this, researchers have experimented with automatic monitoring approaches. Some of the automatic shopping activity monitoring approaches utilise physical items like the shopping cart [51] which is instrumented, or utilise either one or more amongst vision based approaches [58], RF approaches [33] or wearable based monitoring approaches [122]. Since my work utilises wearables and RF sensing, I present relevant work for these categories along with some vision based approaches. Most purely vision or purely RFID based approaches are developed on the server side, with little or no deployment on the customer's devices, while most wearable based approaches require application installation on customer's device. Even though customers have to install custom applications, these applications raise less privacy concern as compared to video monitoring (e.g. [111]). Moreover, applications such as [47] suggest that customers install custom applications, if adequate incentives are provided.

Video based shopping analysis has been explored by several researchers. A Kinect-based system for assessing shopping related actions was proposed in [107]. Using the silhouette data from Kinect, collected in a controlled environment, they identified if the person is examining or picking an item, trying on an item and interacting with the shopping cart, but did not focus on recognizing the actual item of interest. While the authors in [107] used the Kinect for analysis, the authors in [165] used the video data to understand the influence of groups on shopping. From the video, the researchers extracted features which influence shopping – e.g., frequency of touch, path taken etc. Similarly, utilising the video feed to identify not only the path taken but also to identify opportunities to make a sale was suggested in [109]. These studies demonstrate that shopping activities can be identified by

video analysis. However, other than privacy, video analysis also has the problem of occlusion. Shopper's activity might not be captured by the camera and thus video analysis might not produce desirable outcome.

Alternate to video analytics, researchers have looked at purely RF based monitoring. An RFID-based system to infer comprehensive shopping activities like picking item, putting in basket etc. was demonstrated in [135]. However, since the system does not use information from personal devices, the system cannot create an individual-level shopper profile as it does not capture a person-wise item correlation. An alternate RF approach which can provide customer based tracking is to use the Wi-Fi signal from a customer's smartphone. A framework for understanding a shopper's overall *in-mall* movement pattern using smartphone sensors and store-recognition using Wi-Fi was put forward in [67]. In this work, the researchers performed a client side indoor localization to analyse the shopper's trajectory in the mall to understand the malling behavior. In contrast, the Channel State Information of Wi-Fi signals to infer a shopper's locomotive state & location within a store was demonstrated in [164]. This approach identified non-gestural shopping activities – e.g. customer is observing promotions, without any application on the personal device. Not only researchers, but commercial entities are also looking at shopping behaviour; Euclid Analytics [33] capture and analyze the in-store movement of individual consumers by sensing their smartphone Wi-Fi transmissions. Even though Wi-Fi based approaches can assist in identifying customer specific in-store activities, it cannot identify finer activities such as whether shopper picked any items.

Other than RF and video analytics, researchers have analysed sensor data from a shopper's personal mobile and wearable devices to determine the shopping activity. The inertial sensor data to identify shopping ADL was explored in [162]. In this work, the authors divide the user's inertial data (accelerometer and compass) into motifs and determine the shopping activity based on the motifs. However, similar to Wi-Fi based shopping detection, even this approach cannot identify fine grained shopping activities such as whether the shopper is picking an item. To understand

Study	Technique	Store Side Deployment	Personal Devices	Personal Device Sensor	Number of Personal Devices	Approach	Study Location	Study Size	Customer Identification	Pick Identification	Item Identification
I^4S	Multimodal	BLE beacons	Smartphone	Inertial	2	Smartwatch based gesture recognition and BLE based localization	Stationary Store	31	Yes	Yes	Assisted
[67]	Wi-Fi	Wi-Fi AP	Smartphone	Wi-Fi	1	Wi-Fi Based Trajectory	Shopping Mall	195	Yes	No	No
[107]	Video	Kinect	None	None	0	Silhouette analysis from kinect data	Controlled Environment	5	Yes	Yes	Yes
[109]	Video	CCTV	None	None	0	Video Analysis	In-Lab	10	Yes	Yes	Yes
[118]	Inertial	None	Smartphone Smartwatch	Inertial	2	Smartwatch based gesture recognition and smartphone based location inference	Supermarket	25	Yes	Yes	No
[122]	Multimodal	None	Smartphone Smartglass	Inertial Camera	3	Smartphone based location identification, smartglass based gaze detection	Supermarket	7	Yes	Yes	Yes
[135]	RFID	RFID reader and tags	None	None	0	RFID phase change	In-Lab	10+	No	Yes	Yes
[164]	Wi-Fi	Wi-Fi AP	Smartphone	Wi-Fi	1	Wi-Fi CSI analysis	In-Lab	3	Yes	No	No

Table 2.3: Comparison of Various Shopping Activity Recognition Techniques

finer details of shopping, a combination of sensor data mined from a smartphone and a smartwatch to recognize item-level gestural interactions and overall in-store activities was studied in [118]. While the smartphone could identify whether a person was in-aisle or not, the smartwatch could identify the finer details like picking item, putting in trolley etc. However, the system did not try to identify the exact item picked. Tracking different elements of physical browsing such as *dwelling*, *gazing*, *reaching out action* etc. using images, inertial sensors and Wi-Fi data captured from a smartglass and a smartphone was explored in [122]. By analysing the smartglass' images to identify the item that an individual's hand is picking, the system can determine the item of interest. However, other than privacy concerns, continuous image capturing through personal devices can lead to quick battery drop.

In addition to shopping activity recognition, since my work utilises indoor localization, I next describe some fine grained indoor positioning techniques.

Indoor localization: Several indoor localization techniques using Wi-Fi, sound or bluetooth technologies have been proposed by researchers [9, 113, 152]. However, since I needed very fine grained positioning, I have used BLE based positioning in my work, and thus, I will limit the survey to BLE based localization. A BLE-based object localization system which requires at least four BLE receivers to localize an object to which a BLE tag is attached was proposed in [129]. Alternately, fingerprinting-based approaches for BLE-based indoor location tracking of stationary and moving users was studied in [34, 119]. Since we intended to use the BLE based localization in a fixed store layout, we relied on the fingerprinting based technique. An alternate approach to BLE localization is through ranging – proximity based location identification. An adaptive ranging technique using inter-beacon measurements was explored in [115] for indoor localization.

2.4 Behavior Recognition

Till now I have discussed techniques to identify and monitor the physical daily life activity – emphasising on eating and shopping. I next discuss the importance of determining the cognitive state and behavior of an individual during a daily life activity – shopping and the possibility of determining various cognitive states of an individual through mobile and wearable devices.

2.4.1 Understanding the Shopping Behavior

Understanding the cognitive state of an individual during the shopping activity has been of interest to researchers in marketing, social sciences and psychology for decades. To understand why people shop, several hypothesis were put forward by the authors in [144]. One of the hypothesis was associated to the emotional state of the individual – bored, lonely, depressed etc. This indicates that identifying an individual's emotional state might be useful in identifying whether the individual will shop or not. In [90], the authors found that identifying the emotion and behavior exhibited by a store visitor was important as it assisted in increasing the productivity of sales people. This study focused mainly on identifying focused shoppers, which is one of the behaviors exhibited by shoppers. By surveying shoppers, in [39], the authors identified several shopper categories based on the behavior exhibited by the individuals during shopping. Some of these behaviors included: *basic shoppers* – shoppers who knew what they wanted, *destination shoppers* – shoppers who were interested in a brand name, *bargain seekers* – shoppers who were looking for discounts etc. Similarly, in [155], the authors identified that customers can be categorised based on decision making styles, which in-turn affected the behavior. This study also found that gender played an important role in decision making styles. Orthogonal to studies which focused on identifying the shopping behavior, the authors in [19] focused on identifying whether an individual was merely browsing in a store without any purchasing need in mind. The authors indicated that there

were several occasions when an individual in a store might just look at items without having the intent of buying. In general, all these studies indicate that shoppers can exhibit several emotional state and behaviors during a store visit and this was affected by several factors like demographics and environmental factors. Some of the behaviors identified were (a) browsing with no buying intent, (b) focused on what item to buy, and (c) confused about what to buy.

A labor efficient approach to determine the store's customer's behavior is by analysing sensor data from the customer's personal devices or by analysing data from the infrastructure sensors. By analysing the data from the customer's smartwatch's and smartphone's sensors, the authors in [118] demonstrated the possibility of determining whether the shopper is in a hurry. Similarly, the trajectory of a shopper inside a store can be utilised to determine whether the shopper had buying intentions has been shown in [108] and [82]. Even though our shopping behavior identification technique – *CROSDAC* also utilises trajectory to determine shopper's behavior, however, in both [108] and [82], the data used to determine the behavior is from a video feed, which can raise privacy concerns for the shoppers. For *CROSDAC*, we utilised inertial and Wi-Fi scan data of the smartphone to determine the shopper's behavior. A major disadvantage of a video monitoring approach is that it will not be possible to determine the action performed by a shopper in case the video is occluded by objects or other shoppers. Alternately, companies like [33] utilise just the Wi-Fi information to identify various attributes and behaviors of the shopper. Unlike our technique, these approaches ([33, 82, 108]) do not require any software installation on a customer's device.

2.4.2 Automatically Determining Emotions and Behaviors

To identify the emotions and behavior exhibited by individuals, several techniques and approaches have been proposed by researchers. In this section, I focus only on techniques which utilises smartphones, wearables or infrastructure devices.Nudging

the users to manually report their emotional state has been demonstrated in [93]. The researchers analysed the data and provided meaningful feedback to the participants. Alternately, automatic identification of the emotional state of a smartphone user by analysing the user's speech has been studied in [117]. The authors demonstrated the possibility of automatic emotion identification using smartphone's microphone data. Similarly, the use of microphone for deciphering a person's stress state has been reported in [80]. However, a major challenge in the use of microphone is in ensuring that the correct voice is used to determine emotion. In case the phone identifies the voice of someone else in proximity and determines the emotion, then the identified emotion might not be accurate. An alternate automatic emotion identifying approach using the smartphone, by analysing the smartphone's app usage has been demonstrated in [73]. Since the smartphone is usually personal, the possibility of confusing the user's emotion with someone else's is lower than analysing the microphone data.

Other than smartphones, researchers have utilised various wearable devices to understand the human emotion. The possibility of understanding emotions through the measurement of skin temperature, heart rate and electrodermal activity has been demonstrated in [55]. For this study, the authors attached sensors on the subject's skin, but argued that the sensors could eventually be implemented in a smartwatch. Through the study, the authors demonstrated that it was possible to build a psychological database with data collected from multiple participants, which could be used for building a person independent emotion recognition system. Similar to the study reported in [55], the authors in [75] also collected various physiological signals. However, instead of attaching sensors directly to the body, the authors utilised an off-the-shelf device for the sensor data collection.

The previous techniques utilise the change in voice or physiological signals to determine emotions. It is well known that facial expression are a good indicator for several emotions. The possibility of using neurofuzzy networks to determine emotions has been reported in [50]. This work showed that facial expression based

emotion recognition systems can be built without utilising any personal devices. In case of shopping, the CCTV in the store can be used to capture images of the face, which in turn can identify the emotion. There are various other similar studies which utilise images/ video for emotion detection. The possibility of fusing the data from a video feed and EEG sensor attached to an individual to determine the emotion in real time has been demonstrated in [138].

Chapter 3

Monitoring Activities of Daily Living (ADL)

As discussed in Chapter 1, there are various possible approaches to identify and analyse daily life activities. In this chapter, I first explain the need for identifying not just the daily life activities, but also the fine-grained details of these activities. I also describe the approaches applied in this dissertation for this identification. This is followed by a description of some motivating scenarios which demonstrate the usefulness of fine-grained activity monitoring. Finally, I introduce and provide a high level overview of two separate fine-grained ADL monitoring systems/techniques (*Annapurna* and I^4S), and explore an approach (*CROSDAC*) to determine cognitive state of an individual during a daily life activity.

3.1 Identifying and Understanding ADL

To monitor a daily life activity, it is essential to correctly distinguish the relevant marker associated with the activity (e.g. hand-to-mouth gesture might indicate eating or smoking). As indicated in the previous chapter, several approaches have been proposed to determine various user activities [44, 62, 96, 104, 145], which in turn can determine ADLs. An increasingly popular mechanism to identify these

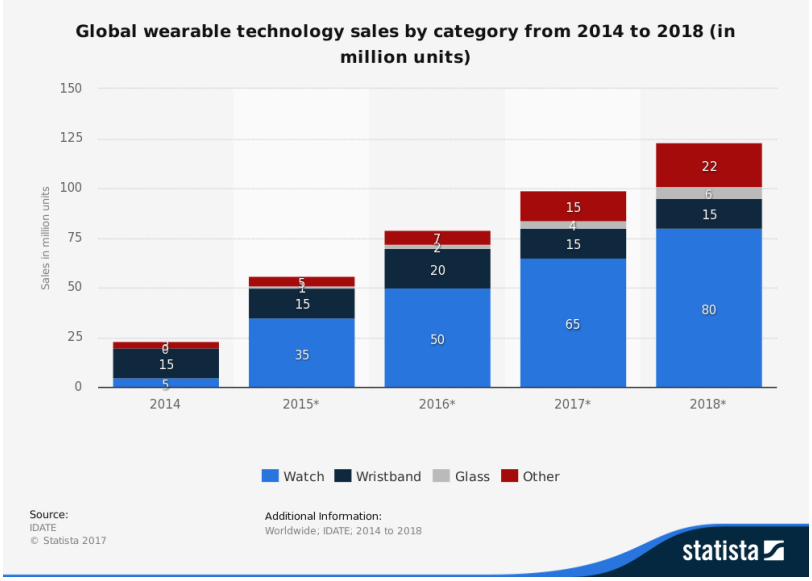


Figure 3.1: Breakdown of Global Wearable Sales

ADLs is through the use of wearable devices (smartwatches [104, 145], smartglasses [88, 122] or other wearables [114, 121]). Based on statistics released by Statista [141] from the DigiWorld Yearbook 2015 [48], the estimated number of wearable devices that will be sold in 2017 is close to 100 million, with smartwatches getting the lion's share of the sales. (Figure 3.1 shows the breakdown of sale based on category.) With such a high penetration of wearable devices (esp. smartwatches), along with the availability of ready to use activity recognition APIs (e.g. Android's Activity Recognition API [6]) and various machine learning techniques, innovative ADL monitoring applications are gradually materialising.

As shown in [104] and [145], analysing data from a single class of sensors on wearable devices can provide details about a specific activity or gesture related to an ADL. However, a more comprehensive and fine-grained understanding of ADLs requires a fusion of sensor information from multiple classes of sensors. These sensors can be embedded in either one device or can be from several devices. To understand this better, let us consider this example: while Stan consumes his meal in the food court, a smartwatch might be able to identify the eating gestures and might be able to determine if he is eating fast, how many spoons did he consume etc. However, there are other contextual informations - e.g. what is Stan eating? is he

sitting while eating? etc. which will be difficult (not impossible) to identify without additional sensors (or devices). These contextual information can provide various analytical insights – for example, a researcher might be interested in identifying the effects of standing while eating on weight gain.

In the subsequent sections, I discuss about some motivating scenarios which will highlight the importance of using multiple sensor classes for fine-grained daily life activity monitoring.

3.2 Motivating Scenarios

3.2.1 Scenario 1

Scenario: Alice, a University freshman, has moved to a new city three months ago, and has been living alone since then. It is her first time alone and she feels it is a huge challenge to keep a healthy eating habit, which was easier previously, when she was living with her parents. She often eats high-calorie food late at night and skips breakfast on most mornings. As a result, she has gained several kilos in the last few weeks. She has also been experiencing heartburn, which led her to visit a doctor.

Possible Solution: For Alice, an automated and unobtrusive diet monitoring system which can capture images of her food plate automatically will be useful because firstly, the system can automatically identify every individual meal that she has consumed. If the system detects that she is eating late at night, it can guide her to eat as little as possible. Secondly, the application can automatically capture the images of food items she is consuming, which she (or her parents) can review once a day or so, and easily keep a diet diary to trace her calorie and nutritional intakes. Finally, the application can also identify the approximate number of spoonfuls that she consumed ([31] demonstrated possibility of monitoring calorie intake from spoonful consumed) and her eating speed. This detail can be included in the food journal.

Possible Steps: To realize the above-mentioned scenario, the diet monitoring system should be able to perform the following:

- *Smartphone* + *Infrastructure*: Identify that Alice is in a location where she might consume food.
- *Smartwatch*: Detect hand to mouth gestures and identify if she is eating or performing some other eating related gestures.
- *Smartwatch*: Determine if she is wearing a smartwatch which has a camera and if the camera can be triggered at the appropriate time to capture an images of her food plate.
- *Smartwatch* + *Smartphone*: Keep track of various eating related analytics like time duration between spoons, count of spoonfuls eaten, is she standing etc.
- *Smartwatch* + *Smartphone* + *Server*: Perform image processing on captured images to determine images in which the food plate is clearly visible
- *Smartphone* + *Server*: At end of day, display the best images of the meal to Alice.

Possible Extension: In addition, the system might identify subtle changes that Alice can consider to her eating style which will help in improving her health – e.g. if the system determines that she is eating too fast, it can guide her to eat slowly so that she might consume less food [83]. This can be provided as a feedback to Alice through her phone. The system might also capture an image of the plate at the end of her meal, using this (together with images taken at the beginning of the meal) to quantify the quantity Alice consumed.

3.2.2 Scenario 2

Scenario: Joey is in the medical professional and usually has long working hours. He gets one off day per week from work, which he utilises for household chores.

One such chore is grocery shopping. Joey maintains a digital grocery list in his smartphone, which he keeps updating through the week. However, during his actual store visit, Joey often gets distracted and forgets to purchase all the items in the list. *Possible Solution:* It will be useful for Joey if his devices could ensure that he purchases all the items in the list. First, it should identify if Joey is in the appropriate store. It should then identify his picking gesture and location within the store to determine what items he is picking and accordingly update his *notYetPurchased* list. Before leaving the store, Joey can glance through the *notYetPurchased* list and ensure it is empty. Since checking for an empty list is less intense as compared to scanning through the entire list and comparing it against items in his basket, it is less likely that Joey will miss out on items.

*Possible Steps:*For this scenario, the item picked monitoring system should perform the following steps:

- *Smartphone + Infrastructure*: Identify if Joey is in the correct shop.
- *Smartwatch:* Determine if Joey has picked an item from a shelf and placed it in his shopping basket.
- *Smartwatch + smartphone + infrastructure:* Determine the rack and shelf from where Joey picked the item.
- *Smartphone + infrastructure:* Access the shop's resource database to determine the item present in the rack. If multiple items are present in the rack, determine the exact point from where Joey picked the item and which item is present at that location. If the item is present in Joey's shopping list, mark that the item has been picked.

Possible Extension: While Joey is shopping, the application might look for offers that might be associated either with the picked item or same items from other similar brands. In case Joey reaches the checkout counter without purchasing any item in the list, then an application on his devices should notify him. The device might look

for checkout queues [99] or infrastructure sensors (e.g. BLE beacons) to determine that Joey is at the checkout counter.

Additionally, in case Joey's smartwatch/phone can determine if Joey is at the checkout counter, it can automatically scan the *notYetPurchased* list and nudge Joey in case the list is not empty.

3.2.3 Scenario 3

Scenario: Penny loves to cook and is also a frequent visitor of the supermarket. She has installed the supermarket's application on her smartphone as well as her smartwatch which provides location based notifications about all possible promotions that the supermarket is currently offering. On normal days, Penny finds this useful. But on certain days when she is in a hurry, the notifications distract her, which in turn delays her further and she finds this annoying. Penny would have preferred an application which could automatically determine when she was rushing and would turn off unnecessary notifications from popping up.
Possible Solution: For this scenario, a sensing application running on Penny's phone should be able to determine her location. On identifying that Penny is in the shop, a combination of her trajectory, locomotion state and hand gestures should be used to determine if Penny is in a hurry. If the devices determine that she is in a hurry, then notifications about promotions inside the shop should not be shown to her.
*Possible Steps:*The scenario above requires:

- *Smartphone + Infrastructure*: Identify if Penny is in the correct shop.
- *Smartphone + Smartwatch* Collect sensor data from Penny's devices to determine fine-grained activities/gestures and in-store location.
- *Server*: From the fine-grained activities and locations, determine the extract Penny's shopping style and match it with historical data from other shoppers to determine the exact style.

- *Server*: Determine Penny's shopping behavior by comparing Penny's shopping style with other shoppers who have exhibited similar shopping style.
- *Smartphone + Infrastructure*: Based on rules, determine if it is appropriate to notify Penny.

Possible Extension: An application running on the supermarket's manager's dashboard identifies the behavior that each customer in the shop is exhibiting. In case the application determines that a shopper appears confused and is looking for assistance, it can alert the store manager, who in-turn can inform sale-assistants present in the store to attend the shopper.

3.2.4 Other Scenarios

The scenarios detailed in this section identifies some possible use cases of fine-grained daily life activity monitoring and of inferring the cognitive state of an individual during an activity. I also discussed some of the possible approaches that may be used to determine the fine-grained details of an activity, by using data from multiple classes of sensors, once the gestural markers of the activity have been identified. The example scenarios are just some possible scenarios where fine grained activity monitoring can be useful. There can be many other similar possible scenarios. Since my work involves eating activity monitoring and shopping monitoring, let me quickly list out some other possible applications which can be built by monitoring these activities:

3.2.4.1 Eating Monitoring

The most popular application of an automated food journaling is for monitoring all food items that an individual consumes through the day. However, there can be various other compelling scenarios where this can be used. Applications related to elderly or child care can benefit from automated food identification. In both cases, if there is a reaction from a food item and the individual cannot express details of all

food items consumed, monitoring the food journal can help in narrowing possible causes of the infection. Similarly, people with mental stress or depression can be identified by analysing their regular eating style (fast or slow) and the surrounding context of an individual (eating alone or in dimly lit locations) over several meals.

3.2.4.2 Shopping monitoring

In the scenarios above, I have described the advantages of shopping activity monitoring applications from a a shop's customer's point of view. However, identifying an individual's shopping activity and understanding the in-store behaviour can be interesting both for the customer as well as the retailer. From an retailer's point of view, the benefits can be in terms of (a) *Managing manpower:* assistance can be provided immediately to shopper's who are looking for items or (b) *In-situ promotions:* offering *on-the-fly* discounts to customers who have a certain product in their trolley. While from a customer's point of view, other than the scenarios above, the benefits can be in terms of *Relevant Recommendations:* get product's location and information that is most relevant.

3.3 ADL Monitoring Systems and Techniques

I next describe three systems/ techniques that I have worked on to demonstrate the possibility of identifying fine-grained context in daily life activity monitoring or understanding individual's cognitive state during the activity.

3.3.1 *Annapurna*: Automated Food Journaling

This dissertation first describes *Annapurna*, a system which has been tailored for a situation similar to Scenario 1 described previously. The *Annapurna* system consists of a smartwatch - Samsung Gear 1, a smartphone - Samsung S III (optional in the system) and a backend server. For the smartphone application, we have also tested it on other Android devices without any glitches.

The goal of the system is to create an automated food journal. To realise this, a custom sensing and processing application runs on the smartwatch. The initial task of the smartwatch is to identify gestural markers associated with eating – hand-to-mouth gesture. For the gestural marker identification, the application turns on the accelerometer sensor to determine eating gestures. Once the accelerometer sensor predicts eating gestures, the application turns on the gyroscope to ensure that the predicted gesture is indeed eating. A two step eating gesture identification was developed because the accelerometer sensor is cheaper (in terms of energy consumption) than the gyroscope sensor. Thus, when there is no eating gesture (most of the time during the day), energy consumption will be low. However, the accuracy of eating gesture determination by an accelerometer is less than the gyroscope. So when the accelerometer detects an eating gesture, it will turn on the gyroscope so that if subsequent eating gestures occur, the gyroscope can filter out false positives. Once the gyroscope determines eating, the camera of the smartwatch is opportunistically turned on and images are captured.Through innovative techniques, *Annapurna* ensures that the image capturing technique is energy efficient. When the smartwatch does not identify further gestures for some time, it switches back to the accelerometer based eating determination mode.

A process running once every few minutes on the smartwatch checks if new images have been captured. If newly captured images are found, they are transferred to the smartphone. The smartphone performs some basic image processing to filter out improper images. Images which might contain the outline of the food plate are sent to the server. A custom image recognition algorithm using opencv [101] running on the server determines the *best* images of the food plate and stores them in the form of a personalised food journal. An individual can log into *Annapurna's* web application at any time to inspect these images.

3.3.2 I^4S: Identifying In-store Interactions

I^4S lays down the steps to realise a system which can determine all the items that a shopper interacted with, while shopping in a brick and mortar store. Such a system would help in fulfilling the requirements of Scenario 2.

The goal of the system is to determine all items that an individual interacts with (picks) during a shopping episode. Even though the smartwatch with multiple sensors can determine diverse contexts, we found that the position of the smartwatch made it incapable of identifying certain fine-grained context (detailed discussion in Chapter 5). We thus designed the I^4S system with multiple devices, with the gestural marker from the smartwatch triggering the entire fine-grained context determination. Devices used to evaluate I^4S includes: a smartwatch – LG Urbane, a smartphone – Samsung S IV, BLE Beacons – Estimote and Wi-Fi access points. I^4S also demonstrates that modestly instrumented environments can assist in finer context determination. The system working of I^4S is as follows: A shopper enters a shop wearing a smartwatch in the wrist and carrying a smartphone. Both the devices record the accelerometer data, the gyroscope data, Game Rotation Vector data along with the BLE and the Wi-Fi scan information. To determine the items that a shopper interacts with, first, using the inertial sensor, the smartphone determines whether the user is stationary (most interactions occur in stationary state). Once the smartphone determines that the user is stationary, the smartwatch looks for the gestural marker of shopping – picking gestures. On identifying picking gestures, the BLE scan information is used to determine the 2 dimensional location of the shopper on the floor plane. Then the watch's inertial sensor determines the hand's location at a shelf level and finally the game rotation vector determines the position in the rack where the interaction took place.

I^4S relies on a backend server to provide item-rack location information. Through a reverse look up, I^4S is able to determine the items that the user picked up during her shopping episode.

System Name	Referenced Chapter	Devices Used
Annapurna	Chapter: 4	Samsung Gear Smartwatch Samsung S III Smartphone Linux Server
I^4S	Chapter: 5	LG Urbane Smartwatch Samsung S IV Smartphone Estimote BLE Beacons Wi-Fi Access Point
CROSDAC	Chapter: 6	Samsung S II Smartphone Samsung S IV Smartphone LG Urbane Smartwatch Estimote BLE Beacon Wi-Fi Access Point

Table 3.1: List of Devices Used in the Studies

3.3.3 *CROSDAC*: Understanding Shopping Behavior

This dissertation explores a technique, named *CROSDAC*, to determine the behavior of a shopper. *CROSDAC* is built to realise scenarios similar to Scenario 3. Devices used in the studies included a smartphone - Samsung S II and Wi-Fi access points.

The goal of the technique is to identify the cognitive state of an individual during shopping. Specifically, *CROSDAC* identifies whether the shopper is confused, focused or has no buying intention. The working of *CROSDAC* is as follows: A shopper enters a store carrying a smartphone. The smartphone captures the accelerometer data as well as scans for Wi-Fi information. At the completion of the visit, these traces are extracted from the phone and analysed to determine trajectory and locomotive features. A clustering technique is applied to this feature set to determine the "shopping style" (a latent attribute) of the shopper. Data from other shoppers who have visited the store previously and have exhibited similar *shopping style* is used to classify the shopper's behavior to determine the shopping-intent. The exploration of *CROSDAC* reveals that *CROSDAC* can assess the shopper's cognitive state even without any personalised training data.

Table 3.1 lists down the chapters where each of the above techniques is explained in details. It also lists all the devices that have been used in the studies.

Chapter 4

Automated Food Journaling

In this chapter I present a system that we have built - *Annapurna*, which not only performs ADL specific gesture identification (eating gesture recognition), but also offers additional informations - a diary detailing user's food consumption, illustrated by the most representative images for each consumed meal. I describe various design considerations and system level challenges that we have addressed while building this system. In short, in this chapter, I demonstrate that it is possible to build an automated food journalling application, by combining the capabilities of inertial and camera-based sensing using commercially available off-the-shelf devices, while addressing various system level challenges.

To realise *Annapurna*, we have built a system which constitutes of three key components: (a) a smartwatch-based gesture recognizer that can robustly identify in-the-wild eating-specific gestures, (b) a smartwatch based image capturer that obtains a small set of relevant images (containing views of the food being consumed) with a low energy overhead, and (c) a smart phone + server-based image filtering engine that removes irrelevant uploaded images. Table 4.1 lists down the devices (and sensors) used in *Annapurna* along with their purpose. In this chapter, I shall first provide details of design choices taken to realise *Annapurna*, then discuss about the evaluation technique and the results obtained for each of the components. Finally I will discuss about the *Annapurna* web application. Let me start

Device	Model	Role	Sensors
Smartwatch	Samsung Gear 1	Identify Eating Gesture Opportunistic Image Capture	Accelerometer, Gyroscope, Camera
Smartphone	Samsung S5	Initial Image Filtering	-
Server	16 core processor, 32 GB RAM, Debian 8 OS	Image Processing Hosts *Annapurna* webapp	-

Table 4.1: Devices Used in Realising *Annapurna*

by providing the motivation of food journaling.

4.1 Need for Automated Food Journaling

Automating the creation of a personal food diary has been a research goal for the mobile sensing community for over a decade. Other than assisting in losing or maintaining target weight, such diaries can capture irregular habits too (e.g. eating too fast or having meals too late at night). To date, proposed solutions either (a) require manual action, as proposed in [125] or (b) rely on specialized wearable sensors and devices such as ear-mounted devices [76], neck-mounted devices [160], specialised gloves [136] or wrist worn devices [3] or (c) need instrumented environments [166]. Such solutions have limited compliance (people fail to faithfully take pictures of every meal & snack), and/or fail to capture both the eating activity and the diet (the food being consumed). Consequently, we explore the development of an automated, completely unobtrusive solution, where a commodity wrist-worn wearable device (e.g., a smartwatch or a smart-band) is used to capture both the eating activity and images of the food being consumed. Motivated by the popularity of smartwatches (with some models such as Samsung Gear 1, Omate Truesmart and Arrow containing an embedded camera – shown in Figure 4.1), our core idea is simple enough: a) the inertial sensors on the smartwatch should be able to identify the eating-related "hand-to-mouth" gestures (similar to 'intake' recognition in [145]); and (b) the embedded camera can then cleverly take appropriate pictures of the food being consumed (when it has a clear, unobstructed view).

However, a practical embodiment of this "simple idea" has three key challen-

(Camera circled in red.)

Figure 4.1: Smartwatches with Embedded Cameras

ges/unknowns: (a) First, can a smartwatch camera plausibly capture meaningful images of the food being eaten? (b) Can the diversity of eating styles, food items and environment be identified through a robust classifier? (c) Even if the image capture is feasible, one cannot indiscriminately video-record the entire duration of all plausible episodes, because of both serious energy overhead and privacy concerns. Can we build an opportunistic and accurate image capturing technique?

To address the above unknowns, we built *Annapurna*– an automated food journaling application. *Annapurna* demonstrates that (a) images of food can be captured in over 90% eating episodes, (b) eating gestures can be captured in real world scenarios with false-positive and false-negative rates of 6.5% and 3.3% respectively and (c) Gesture recognition and Image capturing pipeline can be optimised to capture sequence of images while a person is eating. Next, I describe *Annapurna* and how it addresses the above challenges.

4.2 System Overview

Since eating activity can be highly diverse, we first set the design goals of *Annapurna* and then design the system which addresses these goals.

4.2.1 Design Goals

- *Focus only on **persistent** eating episodes that last at least 5 minutes*: As eating episodes are not fleeting (they last several minutes) and consist of multiple hand-to-mouth gestures, *Annapurna's* eating detection logic need not focus on detecting *every* eating gesture, but can utilize longer observation windows for robustness. We explicitly do not try to track extremely transient eating activities (e.g., picking up a single candy while exiting a restaurant).

- *Focus only on **plate-related** eating episodes*: *Annapurna* can focus on detecting eating episodes that involve some *utensil*–i.e., we do not target scenarios where the user is walking and eating because even though the eating gesture can be identified, it is not possible to capture the image of the food.

- *Judicious image capture and filtering*: To support continuous day-long operation, and address privacy-related sensitivities, *Annapurna* must trigger the camera sensor judiciously, for only short long periods of time and should eliminate images that do not capture the food being consumed.

4.2.2 Overview

Figure 4.2 provides the high-level work flow of the envisioned *Annapurna* system. Broadly speaking, the *Annapurna* smartwatch component must identify the intermittent *eating episodes* during the day, and then trigger the associated camera to capture *likely* images of the food being consumed. Subsequently, these images must be first *filtered* on the smartphone (to remove images that are very likely to not contain any food-related images), and then *ranked* on the server to select a smaller set that best represents the food associated with each eating episode. Finally, these images and other relevant eating-related information should be displayed to the user via a Web-based application.

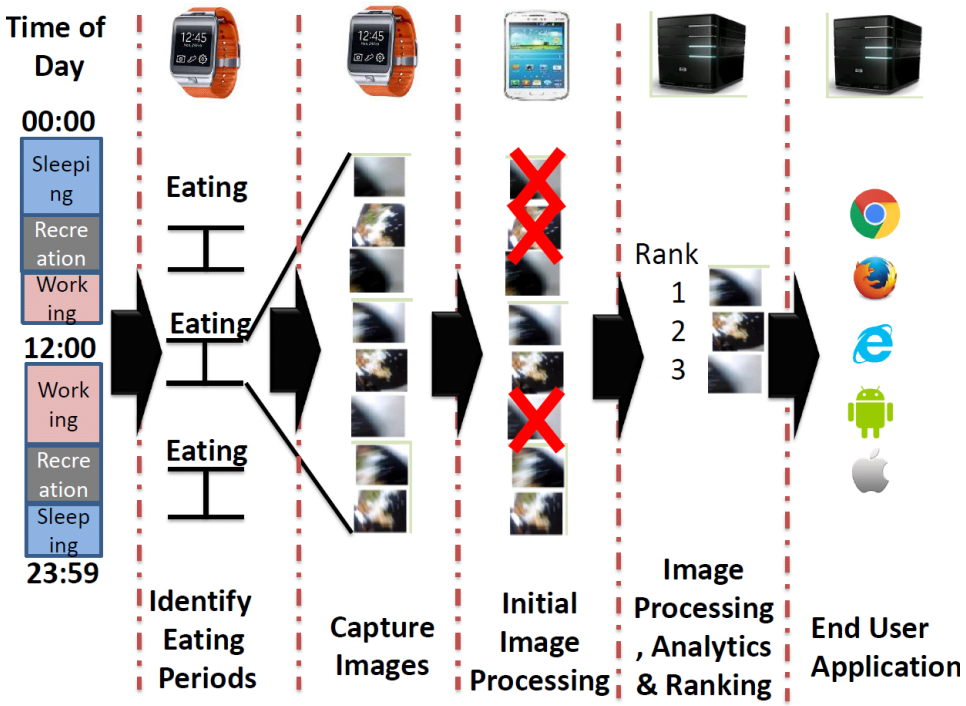


Figure 4.2: System Overview of *Annapurna*

Each of the above steps maps into a distinct technical component of the eventual *Annapurna* application:

1. *Eating Gesture Recognizer:* This module on the smartwatch uses inertial sensors (gyroscope and accelerometer) to detect both (i) the onset of an eating episode and (ii) individual repetitive eating (i.e., hand-to-mouth) gestures within the episode. It should accommodate the variations in the sensors' readings introduced by the diversity of users and eating styles. The classifier must balance the possibly conflicting precision and recall goals: while it should not miss any of the eating episodes, it should also ensure that other similar gestural activities (e.g., washing one's face, cooking) do not result in a false 'eating' classification. As we shall see later (in Section 4.3), we eventually converge on a two-stage classifier that achieves both low energy overhead and lower false-positive rate (higher recall).

2. *Responsive Image Capturer:* Once the onset of an eating episode has been identified (via multiple closely-spaced eating gestures), this module captures

images automatically. Capturing images automatically is challenging because the hand-to-mouth gesture is relatively short (on average about 3 seconds) and the latency of image acquisition by a smartwatch camera (i.e., the time to actually turn on and capture an image) is fairly high (more than 800 msec). In Section 4.4.2, we shall show how our choice of a preview-mode based image capture strategy provides the ability to capture a sufficient set of useful images, while tolerating uncertainty in the precise trajectory of each individual eating gesture.

3. *Image Filter:* This component performs both the steps of (a) irrelevant image elimination, followed by (b) selection of the best set of images for each eating episode. Given that image processing and filtering is a computationally complex process, only simple (but effective) image pre-processing happens on the smartphone, with the bulk of such image analysis being performed on a backend server. In Section 4.4.2.1, we shall derive the detailed algorithms for elimination and ranking, and show how *Annapurna* provides transport efficiency by performing computationally-efficient filtering on the smartphone.

4. *Food Journaling:* Finally, the server stores this small subset of *relevant* images corresponding to each detected eating episode. The user can then view these images at any appropriate time (e.g., once every night) via a Web portal. While the portal development is straightforward, in Section 4.5 we shall discuss some design choices (e.g., number of images per episode to be presented to the user) intended to improve the overall *user experience*.

4.3 Design Choices

With the goals in place, we next discuss the design choices taken while building *Annapurna*. Rome wasn't built in a day; neither was *Annapurna*. The system building was an iterative process where, lessons learnt in a version was used to refine

Food Item	Preferred Eating Modality	# of Episodes	Completion Time (sec)			Hand to Mouth Gestures			Percentage of Episodes with Useful Frames
			min	max	avg	min	max	avg	
Rice ≈100gms with 2 vegs	fork & spoon	66	211	1140	568	22	54	33.5	95.5%
Sandwich (bread/ burger & fries)	hand	20	255	363	299	6	35	14.4	65%
Pasta / Soupy noodles	fork / chopstick	29	234	771	459	13	35	27.3	86.2%
Fruits ≈15 pieces	fruit stick	20	51	387	183	7	23	13.5	70%

Table 4.2: Key Results from Micro Studies

the subsequent versions. Overall, the vision of *Annapurna* required us to address two basic questions: (a) What relevant aspects of real-world eating activities do we need to incorporate in the design of robust classifiers for eating detection? and (b) can a smartwatch camera even plausibly obtain an image of the food being consumed? If so, does this depend on the type of food, and on the on-watch placement of the camera sensor? In this section I describe the design choices and rationale for the system design choices. However, I will first describe the multiple studies that were performed and the lessons learnt in each of the study. Lessons from one study acted as a building block for the subsequent studies and thus will explain all design choices which we have taken.

4.3.1 Micro Studies and Observations

During the course of developing *Annapurna*, several system-level choices were taken based on lessons learnt from previous *Annapurna* versions: an initial prototype was developed based on observations and learnings from a set of *micro-studies* (there were more controlled studies performed before the reported micro-study). In this subsection, details of the dataset and observations is reported.

4.3.1.1 Micro-Study Dataset Details:

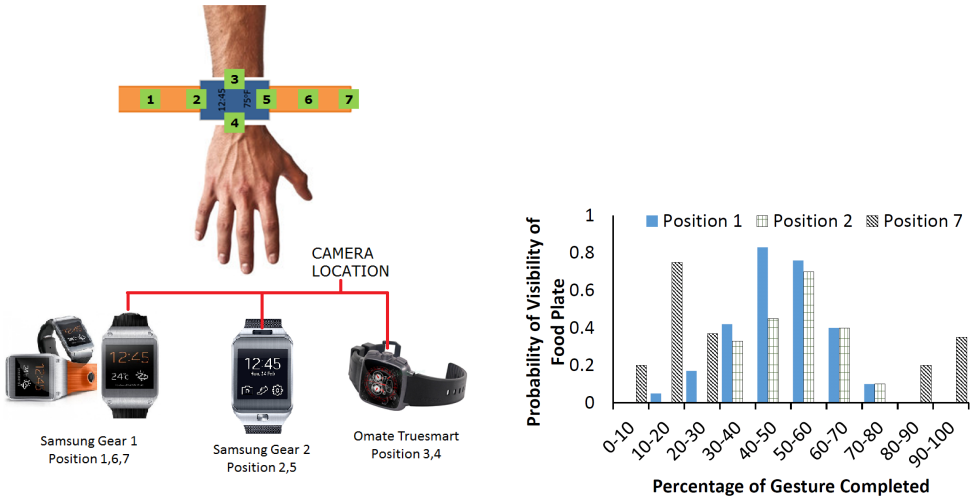
To understand the possibility of meeting *Annapurna's* design goals, we performed a fairly extensive set of *micro studies* with 21 participants (8 females, 13 males,

belonging to 5 nationalities), employed in our university research lab, for a total of 135 eating episodes, where an episode is defined as the period between starting of a meal (after the purchase) and consuming the last spoonful. The meals were consumed during regular food hours when the participants went for lunch, snacks or dinner. Most episodes took place in the university's underground food court (artificial lights), with a few occurring outdoors (natural light) or in covered, open areas (mix of artificial and natural light). During the meal, the participant wore the watch on their dominant hand (all 21 participants turned out to be right-handed). A custom application running on the watch collected accelerometer data, gyroscope data and continuous image frames from the smartwatch during the entire episode, while an external observer video-recorded the meal (for ground truth labeling). Other than the micro study involving 21 participants, *feasibility studies* involving 2 users (both were right-handed South Asian males, one user was 32 years old, while the other was 34.) were also investigated for the sensitivity to the on-body location & orientation of the smartwatch camera.

Table 4.2 highlights some of the key parameters associated with the consumption of these food types; from these studies, we find that there is a wide variation in eating gestures for different food types. They lasted about 3.5 to 20 minutes, involving 13 to 35 separate hand-to-mouth gestures. Among these food items, we also observed that: (a) sandwiches/fruits presented the least number of distinct hand-to-mouth gestures (as users often held the items close to their mouth between successive bites), (b) "noodle soup" had high variability in the number of hand-to-mouth gestures mainly due to the use of forks vs. chopsticks (use of chopsticks, generally leads to higher number of gestures). The variations for "rice" are generally due to the individual's eating speed and quantity consumed in each mouthful.

4.3.1.2 Possibility of Image Capture

From analysis of the images captured by the smartwatch camera in the micro-studies, we obtain the best case (upper bound) on the likelihood of there being



(a) Different Positions of Camera (b) When is the Food Plate Visible?

Figure 4.3: Capturing Food Images vs. Smartwatch Position

at least one image that provides an unobstructed view of the consumed food item. We note that the likelihood of obtaining a usable food image is fairly high (80% or higher), except for sandwiches and fruits (in situations where the user never put the food item down on the plate).

4.3.1.3 Orientation of the Smartwatch Camera:

We also experimented (with the 2 users performing the feasibility studies) with three distinct smartwatches, Samsung Gear 1, Samsung Gear 2 and Omate TrueSmart (illustrated in Figure 4.3 (a)), each with the camera mounted in a distinct position on the outward or inward rim of the watch bezel or on the strap. By varying the orientation on the wrist, we obtained 7 different camera positions (Samsung Gear 1 for positions 1,6 and 7; Samsung Gear 2 for positions 2 and 5; and Omate TrueSmart for position 3 and 4), as illustrated in Figure 4.3(a). For each distinct camera position, the two users consumed one meal each with spoon and fork in the university's underground food court. The users ensured that the watch was not covered and the camera could capture video continuously. On analyzing the captured video, we find that the food plate is visible at least once (for both users) only for camera positions 1,2 and 7 – more specifically, a useful image is found in 82.6%, 77.4% and 80.4% of

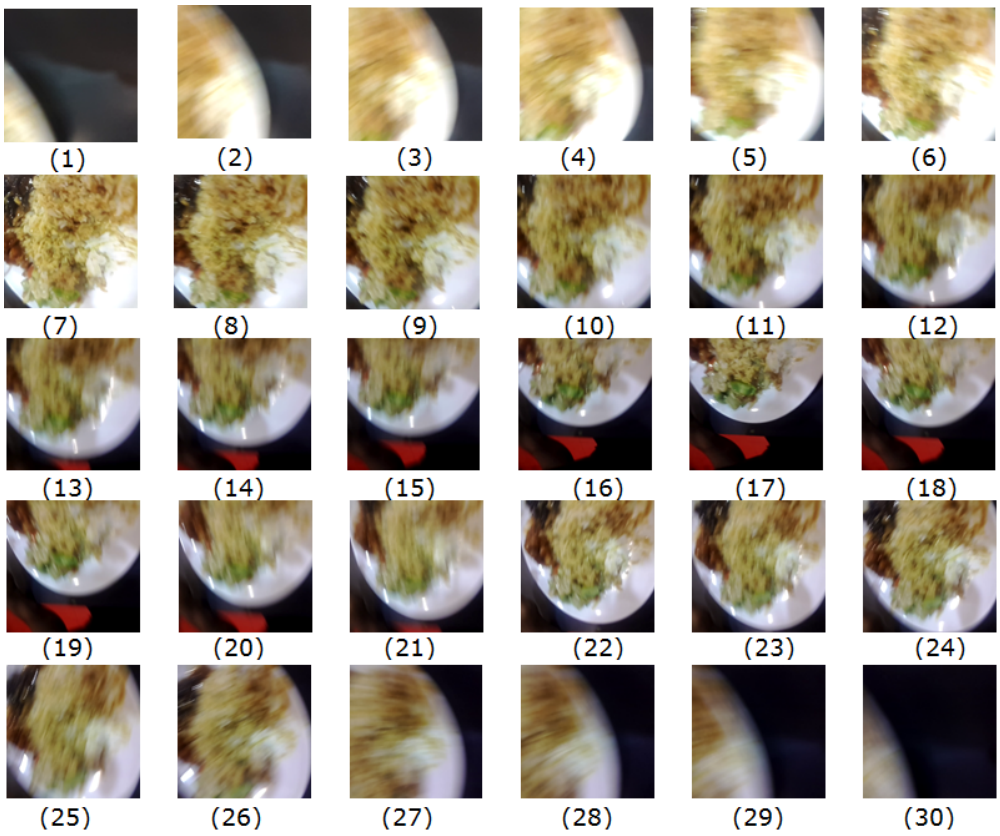


Figure 4.4: Frames Extracted from the Video During a Complete Eating Gesture

all eating gestures, respectively, for these positions. Moreover, Figure 4.3(b) shows the probability of the images being useful (i.e., the food item is visible) as a function of different points in the gestural sequence (the 50% point corresponds roughly to the zenith, where the hand is closest to the mouth). We see that the on-watch camera position significantly affects this probability – for Position 1 & 2, the plate is most visible when the hand was near the mouth.

4.3.1.4 Image Capturing Approach

For the feasibility studies, we had continuously recorded a video. Alternate approaches to video recording is to capture images either in burst mode or in preview mode. In Section 4.4.2, we shall explain the preview mode and further explore the choice of the appropriate mode of capturing images–via continuous video or preview mode. Here, we focus on determining the best strategy for capturing a

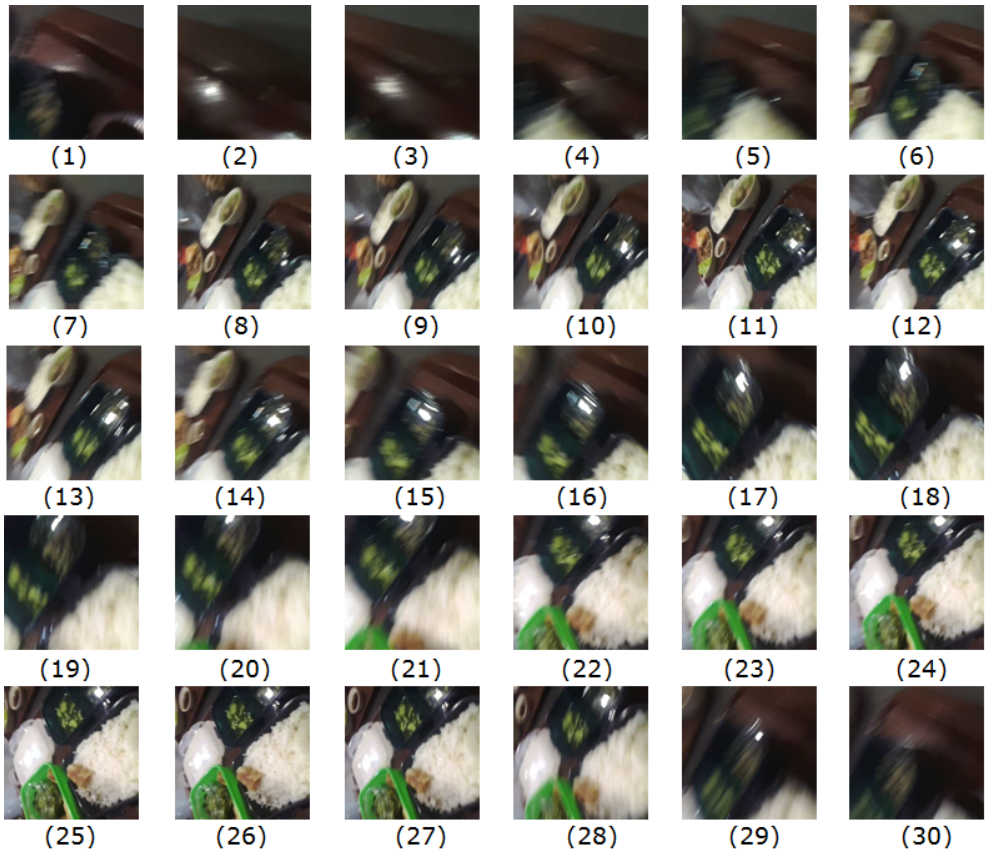


Figure 4.5: Frames Captured in Preview Mode During a Complete Eating Gesture

useful set of images, deferring the discussion on the choice of the best mode to Section 4.4.2. To understand the difference in image quality between images captured in preview mode and extracted from video frames, we extracted the images captured in the preview mode during a micro study and from a video captured during one of the feasibility studies. Both the episodes occurred under similar environmental conditions. Figure 4.4 shows the image frames for an eating gesture, extracted from a video captured by the smartwatch. During the gesture (and episode), the user (Indian male) consumed rice and vegetables from a plate in the university's food court. Similarly, Figure 4.5 shows images extracted from one gesture, where images were captured in the preview mode, while the user (Vietnamese male) consumed rice and meat from a multiple utensils. From the images in the two figures, we can see that the images during the initial and the final part of the gesture are blurry as compared to the images captured around the halfway mark of the gesture.

Figure 4.6: Sample Images Classified as Usable Images by the Two Annotators

Figure 4.7: Sample Images Classified as Not-Usable Images by the Two Annotators

During the two gestures, a large portion of the food plate is visible. However, it must be noted that it is not necessary that every gesture in an episode will capture the image of the plate. During certain gestures, only a part of the plate might be visible, while in others, the plate might not be visible at all.

To answer questions like: *Will the first gesture always capture the image of the food plate?* or *If we capture images till k gestures, how likely is it that we will capture an image of the food plate?*, we extracted the frames captured in all the episodes of the micro-studies. For these frames, we wanted to understand the difference in obtaining useful images for two strategies: (a) *Till-gesture:* if we captured images (or frames extracted from video) continuously till the occurrence of the k^{th} eating gesture, vs. (b) *In-gesture*: if we captured images (or frames extracted from video) till the k^{th} eating gesture, but only when a gesture occurred (and turning image capture off between gestures). Both the approaches had their own merits.

We developed an image recognition and ranking system (explained in Section 4.4.2.1) which identified the P-"best" images from amongst all the images that were captured till the targeted gesture. From P, we picked the top p ranked images (p=1,2,5,10). Two annotators manually inspected the images to determine if at least one manually identifiable good image was present in the top p images

selected for a particular gesture. Figure 4.6 shows some sample images, which both the annotators categorised as *useful images*, while 4.7 were some of the *not-useful* images.

For every gesture (k) in an episode ($i, i \in N$), the value of Good Image(G) is assigned as 1, if atleast 1 manually identifiable image of the food in the plate amongst the p images in the gesture is present.

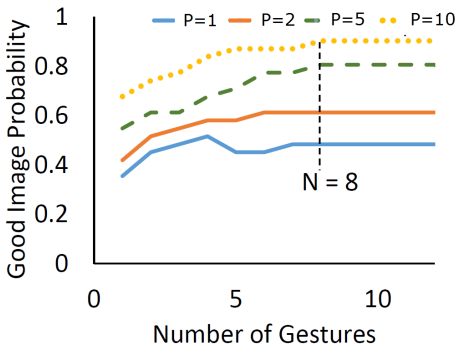
$$G_{ik} = \begin{cases} 1, \text{if gesture} k \text{has good image} \\ 0, \text{otherwise} \end{cases}$$

The *Good Image Probability* (GIP) for gesture k is calculated as

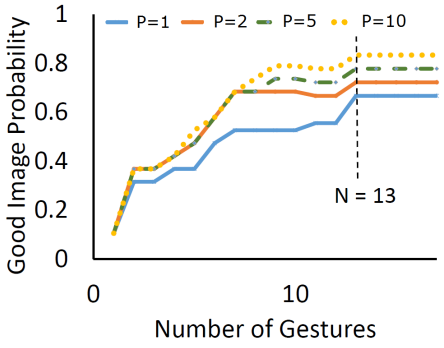
$$GIP_k = \frac{\sum_{i=1}^{N} G_k}{N}$$

For $N = 135$, Figure 4.8 represents the *GoodImageProbability* for images with identifiable food content for the two strategies, as a function of k. From the figure, we see that to achieve a likelihood of having at least one good image exceeding 0.8, the Till-gesture strategy would need only 8 gestures, as opposed to 13 gestures for the In-gesture approach. However, in terms of energy, the Till-gesture strategy would imply that the camera sensor will be used for about 135 sec (average time it took for a user to consume 8 mouthfuls of food). In contrast, the In-gesture approach would keep the camera sensor on for only approximately, $13 * 3.1 = 40.3$ seconds.

To understand if an image recognition algorithm could identify the images which were labeled as *not-useful* images by the two annotators, we passed the images shown in Figure 4.7 (along with other *not-useful* images) to Clarifai, a commercial deep learning based image recognition system. Figure 4.9 shows the images along with the predicted label for the image and the label probability. From the images we see that all the images which were discarded by the annotators as *not-useful* images, obtained a high probability of being a *food* images by the image recognition algorithm. This indicated that even if an image might not be presenta-



(a) Hit Rate when All Images Captured *till–k gestures* are Considered



(b) Hit Rate when All Images Captured *in–k gestures* are Considered

Figure 4.8: Evaluation of Two Strategies to Capture Food Images

Figure 4.9: Image Label Prediction Using a Commercial Image Recognition System for the Not-Useful Images.

ble in a food journal,it might still be useful in identifying the existence of an *eating* episode.

4.3.1.5 Alternate Image Capturing Approach

Currently, we have captured images in preview mode during the micro-studies. Additionally, during the feasibility studies, we captured video from the watch. To understand if capturing a video is more effective than capturing images in the preview mode, we considered two episodes – one from the feasibility studies and one from the micro studies. Both the episodes occurred under similar conditions in the underground food court. In both the episodes, the participant consumed noodles

with a fork. The episode where the video was captured had a duration of 4 minutes and 24 seconds, while the episode where preview mode data was captured was 5 minutes 40 seconds long. We compare the two modes in terms of total image size, power consumption and quality of capture.

Total Size: The size of the video file (video frame size was 640 x 640, while video was captured at 16 fps) was 308 MB, while the total size of all the preview mode images for a longer episode was only 35 MB. If all images were to be transferred to the server, the number of bytes transferred for the video mode would be $\approx$10 times more than preview-mode. The larger size of the image indicates that the image of the food plate might be clearer in the video mode images, which might improve the identification accuracy. As we shall see later, if we capture the first 30 seconds of this video, we can capture a reasonable image, For the preview mode, 45 seconds of images provided a good image.

Power Consumption: To understand the power implications of capturing images in preview mode or recording a video, we measured the energy consumption during each of the mode using a Monsoon Power Monitor [92]. From the measurements, we found that the average power consumption in video mode is 200 mW higher than the preview mode (Figure 4.15 shows the power consumption by different strategies). While the average power consumption in the preview mode was 813 mW, the average power consumption of the video mode was 1021 mW. However, since the smartwatch was attached to the monsoon power monitor during the power measurements, the power consumption reported might vary in real world environments, where the scene captured by the camera might continuously change and there might be additional power consumption due to auto-focusing, changing light color on sensor, etc.

Image Quality: Finally, to understand if the images captured in the video mode were better than the preview mode images in terms of item identifiability, we passed the preview mode images as well as images extracted from the captured video to Clarifai. For both the modes, the first hand to mouth gesture occurred at approxima-

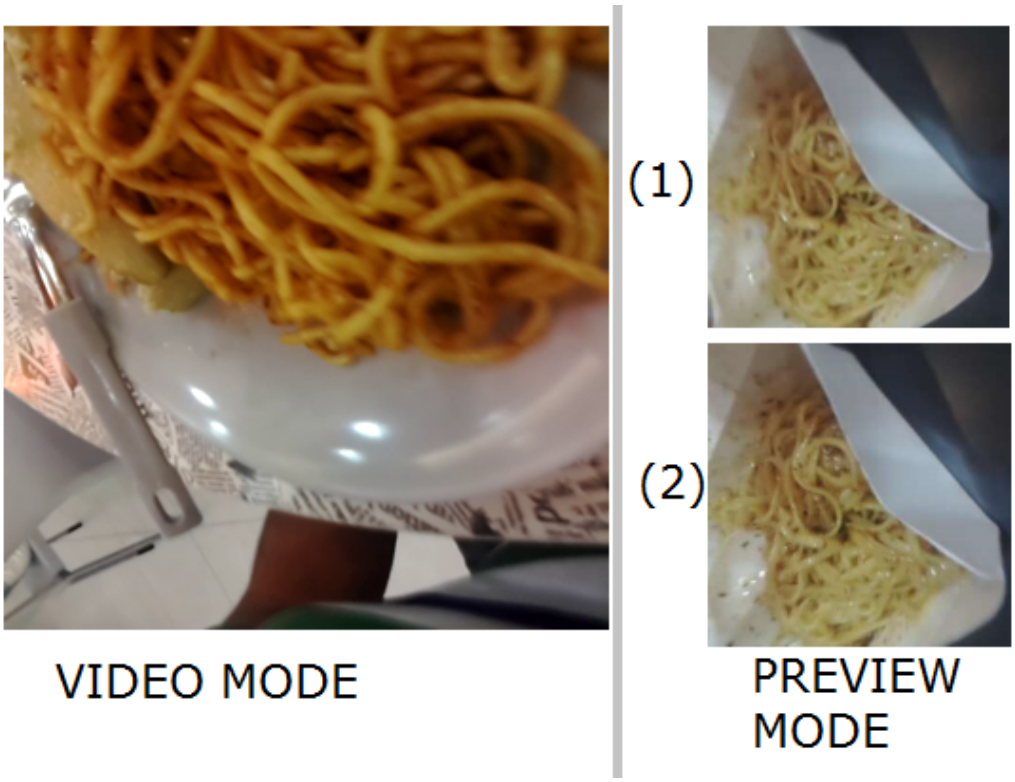


Figure 4.10: Images Captured in Preview and Video Mode. (Scale 1:2)

tely 25 seconds from the commencement of the episode. The occurrence of the first gesture at 25 second was coincidental and it is not necessary that the first gesture will always occur at 25 seconds for other episodes.

Every image captured from the start of the episode upto the end of the first hand-to-mouth gesture was passed to Clarifai. The images shown in Figure 4.10 shows the 1:2 scaled images from the two capturing mode. For the preview mode, '(1)' is an image from the first gesture, while '(2)' is an image from the second gesture. The images shown in the figure are the ones which obtained the best prediction probability. For the video mode, the top predictions along with the prediction probability were: (i) food(0.993), (ii) spaghetti(0.989), (iii) pasta (0.989) and for the preview mode, the top predictions were: (i) food (0.991), (ii) herb (0.971), and (iii) meal (0.944). From the top three predicted items, we can see that for the image captured by the video, the software could determine that the food item being consumed was indeed spaghetti. However, for the preview mode, even though the system identified that food was present, the actual food item prediction probability was low (the

system determined the item to be pasta with a probability of 0.869). We thus discarded this result and passed subsequent images upto the next gesture to Clarifai. The second gesture for this episode occurred at 45 seconds. '(2)' in Figure 4.10 represents the image with the best prediction probability. The top prediction probabilities for the image were: (i) food (0.995), (ii) herb (0.984), and (iii) pasta (0.957). This reasonably high prediction probability of pasta indicated that the system could identify the image.

Even though the comparison shown here is done with just two episodes, however, the images classified during the gestures well represent images that are usually captured in similar episodes. From this simplistic comparison, we found that the prediction probability for the correct item is more likely in video frames, but the size of the video frame is larger and capturing a video requires more energy. We thus used the preview mode in our studies.

4.3.2 Choices That Did Not Work

Based on the observations in the micro-study, we built an initial prototype of *Annapurna*, where, (1) the smartwatch identified eating gestures, captured images during the hand-to-mouth gesture and transfer the images to the server and (2) server performed the image processing and stored the images. While building this prototype we observed/learnt the following

We had issues with capturing images during the in-gesture period. The entire eating gesture lasts for approx. 3 seconds. We broke the gesture into segments and turned on the camera to capture video when we identified the first segment - initial rising of hand. In most cases, the video captured the descending phase of the hand - coming back to position of rest. The reason for this was – there was latency in identifying the gesture along with latency involved in turning on the camera, showing that the in-gesture technique could capture only images from the descending phase of the

hand. This was not identified in the micro-study phase because we were capturing a continuous video in the micro-study and these latencies did not cause an obstruction.

Based on power measurements, we found that energy consumed in capturing an image was lower than that of a video. We thus wanted to capture a burst of images when the user was eating. However, we found that for every image, the camera would adjust focus to capture a good quality image and after an image has been captured by the camera, the camera was turned off by the OS. For *Annapurna*, we found that even though we had looped the image capturing callback to capture the next image, there was almost a 900 ms difference between capturing subsequent images. This indicated that during an eating gesture, we would capture 3-4 images and it was not necessary that they would be useful. On investigating further, we found an alternate technique that represents a compromise between these two extremes– preview mode (explained in Section 4.4.2), which we have used in our prototyping.

From the micro study data, we had tested various classifiers - whether they could identify eating gestures. The result for this is reported in Section 4.4. However we found that in real world, a cross validated classifier did not work well (explained in Section 4.3.3 – lack of training data). So, to improve the design we tested other classifier options such as a single class SVM classifier. However, a single class SVM did not improve the eating gesture recognition accuracy. We thus had to resort to in-the-wild data collection for non-eating gestures.

4.3.3 In-the-Wild Studies

This subsection the in-the-wild dataset and lessons learnt in each study. For the in-the-wild studies, we asked participants to wear a smartwatch (with *Annapurna* running) on their dominant hand and continue with their daily tasks. In each of

User Study	(Users,) Duration	Eating Detector	TP	FP	FN	Image Filtering	Per User Daily Upload	Problems
1	7 users, 5 days	Simple Classifier	31	60.3%	0%	Server	24836k	High-false positive, high energy overhead
2	6 users, 2 days	Cost-based Classifier	11	0% (31.3%)	35.3%	Server	14546k	High-false negative, high data cost
3	4 users, 5 days	2-stage Classifier	29	6.5% (23.7%)	3.3%	Phone & Server	16226k	-

TP=true positive – eating episode was correctly detected, FP=false positive – eating episode falsely detected,

FN=false negative – eating episode missed.

(Numbers in bracket in FP column indicates false positives before the image filtering step)

Table 4.3: Details of In-the-Wild Studies for *Annapurna*

these studies, the participants separately manually recorded the ground truth (what they ate, when and how–i.e., with chopsticks, forks, etc.). Naturally, the eating activities spanned a wide variety of environments (restaurant, in a movieplex, food court, outdoors, at home), and involved various types of utensils, sitting position and lighting arrangements. Table 4.3 provides a succinct summary of issues observed for each of the technical components (if any) from each study, and how the component was refined to overcome these issues.

Study 1: A total of 7 participants (4 females, 3 males) were recruited from our lab and were asked to register with *Annapurna*. They were provided with the watch (which they were instructed to wear in their dominant hand) and the phone. They were also asked to appropriately recharge the battery whenever it drained out. There was no requirement laid regarding meals to eat and places to eat. Other than this, the users were also asked to validate the accuracy of the system at the end of the day and to ensure that they approved all the images that were uploaded.

By day 3 of the study we found that our gesture recognition system had high false positives, leading to large volumes of data being upload to the server and rapid drainage of the smartwatch battery. (Nonetheless, the participants used this version for 5 days, capturing a total of 31 eating episodes during periods when the watch had sufficient battery.) This problem was traced to our use of a very lightweight classifier (chosen to ensure it could run on the watch) and the lack of robust *real world* data of a variety of non-eating activities. Consequently, we eventually

Id	True Positives	False Positive Rate	False Negative Rate
Participant 1	9	10% (31%)	10%
Participant 2	7	13% (22%)	0%
Participant 3	7	0% (22%)	0%
Participant 4	6	0% (15%)	0%
Overall	29	6.5% (23.7%)	3.3%

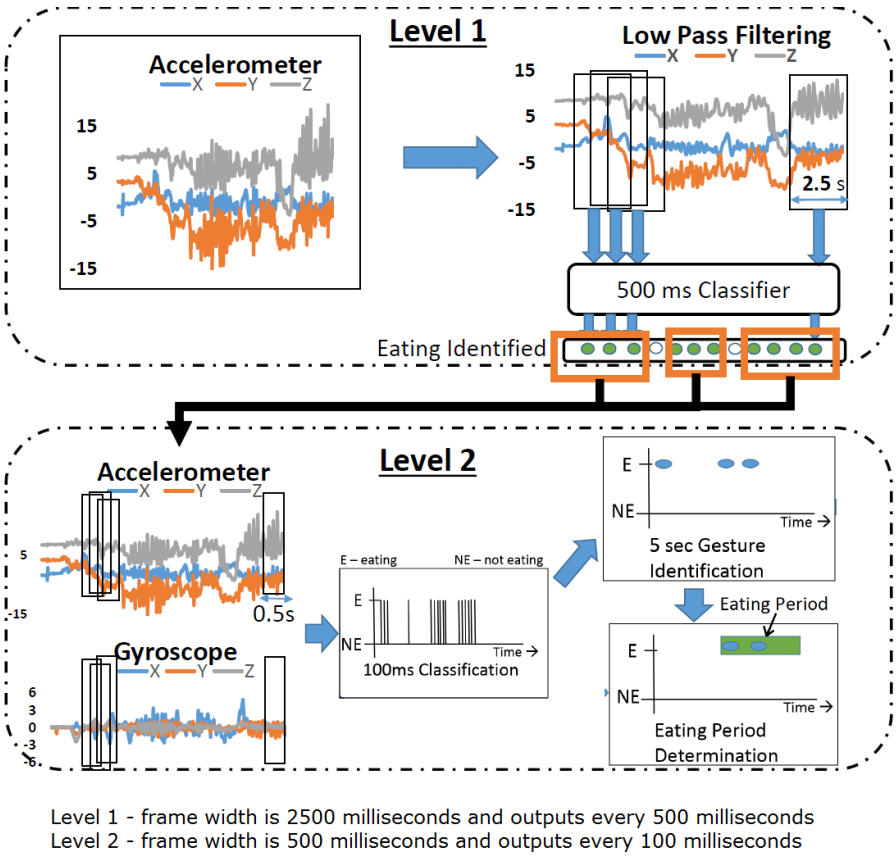
(Numbers in bracket in False Positive Rate column indicates false positives before the image filtering step)

Table 4.4: System Performance for Each Individual Particating in the In-the-Wild Study:3

switched to a cost-based classification approach, where false-positives were more heavily penalized.

Study 2: We then redeployed the improved system (with a cost-based classifier) on 6 users (one of the original users dropped out) and evaluated it for 2 days. The new system significantly lowered the false positives in gesture recognition (we had only 5 false positives for eating generated by the gesture recognizer; all of these cases were eventually filtered out (by the image filtering step) as they contained only irrelevant images). However, this classifier now exhibited higher false negative rate– we missed out 6 eating episodes over those 2 days. To subsequently tackle this issue, we then developed a two-stage eating detection classifier (details in Section 4.4.1.4).

Study 3: The final refined version of the *Annapurna* client was deployed to 4 (out of the original 7) users over another 5 day period. In this study, 29 eating episodes were correctly identified and images were accurately captured. There was 1 eating episode which the system could not identify. In terms of false positives, there were 9 false positive episodes, where even though no eating took place, the eating gesture recognition model determined that the individual was consuming a meal. Out of the 9 episodes, 7 were filtered by the image processing algorithm. 2 episodes were falsely shown to end users as eating episodes. Table 4.4 summarises per-individual performance of the system. From the table we can see that even at individual level, the system performs reasonably well. There were just 2 false positives (1 each for participants 1 and 2) and 1 false negative (for participant 1). Using this study, we



Level 1 - frame width is 2500 milliseconds and outputs every 500 milliseconds
Level 2 - frame width is 500 milliseconds and outputs every 100 milliseconds

Figure 4.11: Eating Period Recognition Approach

were finally able to demonstrate our target goal of achieving both low false positives and false negatives.

4.4 Methodology & Results

We now describe the 3 components, i.e., (i) Eating Gesture Recognizer, (ii) Image Capturer and (iii) Image Filter.

4.4.1 Detecting Eating Gestures

Identifying eating gestures, using accelerometer and gyroscope sensors, has been studied in the past by various researchers [133, 145]. Our overall design of the classifier for detecting eating (both an eating episode and its constituent multiple 'hand-to-mouth' gestures) is shown in Figure 4.11. The entire process can be divi-

Classifier	Accuracy	Precision	Recall
Decision Tree	96.63%	96.1%	96.5%
Random Forest	98.19%	97.1%	99%
SVM	85.66%	83.6%	87.1%

Table 4.5: Accuracy in Identifying Eating Gestures

ded into four parts. We first describe the initial implementation of this classifier, and then separately describe the refinements that we made based on our user studies.

4.4.1.1 Feature Extraction and Low Level Classification

We extracted the raw accelerometer and gyroscope data from the eating episodes and from the ground truth file, marked the period where the eating gesture occurred. From this data we found that an eating episode, on an average, has about 18 to 19 eating gestures. Our initial approach was to use features defined over short frames of both accelerometer and gyroscope data. The small frame size is needed to trigger the camera reasonably in advance to get appropriate images of the food plate. This approach is shown in the bottom part (Level 2) of Fig 4.11. The raw sensor data is partitioned into frames of length 500 msec (with 80% overlap between frames); a set of widely-used features (identical to [159]) are then derived for each frame. These features included both time domain features − mean ($\bar{x}$, $\bar{y}$, $\bar{z}$), magnitude ($\sqrt{(x^2 + y^2 + z^2)}$), variance ($var(x)$, $var(y)$, $var(z)$), covariance ($covar(x, y)$, $covar(y, z)$, $covar(x, z)$), as well as frequency domain features − Energy and Entropy from both accelerometer as well as the gyroscope. From the features we built *person-independent* classification models. We performed a 10-fold cross-validation using a Decision Tree, a Random Forest and a SVM classifier using Weka [41].

Table 4.5 shows the accuracy of identifying eating gestures for three popular classification schemes. While both the Decision Tree and the Random Forest classifier offer high classification accuracy, we selected the Decision Tree classifier (for watch-based gesture recognition) due to its lower computational complexity.

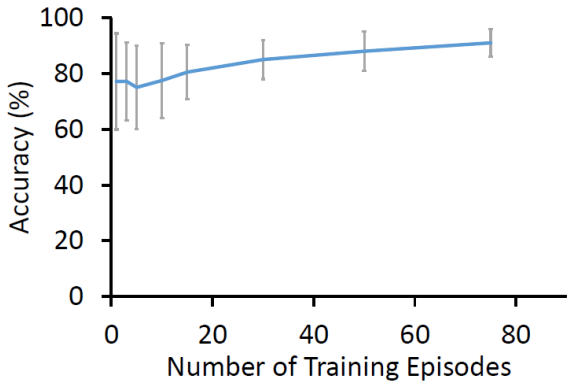


Figure 4.12: Variation of Accuracy as Training Data Size is Varied

To understand the dependence of performance of an eating model on the size of the training data, we took the 95 rice and noodle episodes from the micro-studies. To analyse the performance of each episode, we selected $n : n \in \{1, 3, 5, 10, 15, 30, 50, 75\}$ training files. The training files were selected randomly from amongst the other eating episodes. This process was repeated 10 times with 10 seeds for the random number generator. Figure 4.12 shows the overall average performance for different values of n. From the figure we can see that for $n = 1$, the prediction accuracy is 77%, which appears to be reasonably high. However, on scrutinising the prediction results, we found that for 36% of the experiments, every instance in the episode is predicted as not-eating. There are several reasons for the high false negative rate: (a) Since the data has episodes where either rice or noodles were consumed, if the training happens with an episode where rice was consumed and the tested episode is noodles, or vice-versa, in such a case it might be difficult to predict the eating gesture (b) As participants ate the food using various combinations from spoon, fork and chopstick, with a single training episode, it is unlikely that both the training and testing episode had the same eating modality. (c) There can be high inter-person diversity when only one episode is used in training. The eating style of the person whose data was used for training might not be similar to the person whose data is tested.

The percentage of episodes where all instances are predicted as not-eating drops to 10% when we use 15 episodes for training and further to below 5% when 30

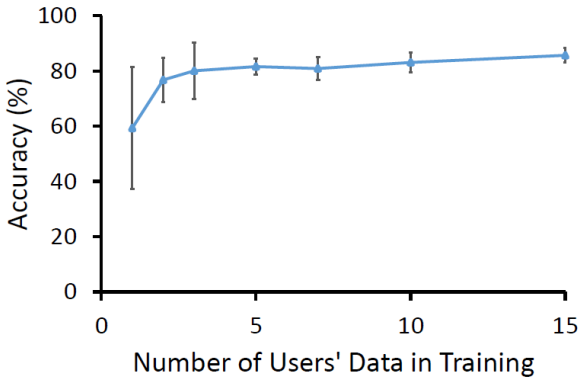


Figure 4.13: Variation of Accuracy as Number of Users is Varied

episodes are used for training. At the same time, the prediction accuracy when using 30 episodes for training is above 85% even when a person-independent classifier is used.

We next analysed the variation in accuracy when all episodes of p : $p \in \{1, 2, 3, 5, 7, 10, 15\}$ participants was used to create the training model and it was tested on the remaining participants. For the analysis, we again used the 95 episodes from the 21 participants, where either rice or noodle/pasta was consumed. Since every participant consumed both rice and noodles/pasta, every training model created had diversity in terms of food consumed. For the training data, we randomly selected p participants from the pool of 21 participants. For every value of p, the process was repeated 10 times. Figure 4.13 shows the variation in performance when the value of p is varied. From the from the figure we can see that for lower number of users in the training data, the accuracy of the system is low. This gradually stabilises as more user's data is user in training. From the figure, we also see that there is significant variation in performance at lower values of p. This indicates that the performance of the system is affected by the users selected for building the model. However, for $p >= 5$, the variation is low. This indicates that for this dataset, data from 5 participants is sufficient to make the model robust.

When using the classifier on 500 msec windows of sensor traces, we found that even during an eating gesture, two consecutive frames were not always classified as *eating*. There were also periods during the eating episode when *non-eating* gestures

(adjusting one's hair, raising the hand to wave at a friend, etc.) were classified as eating in various 500 ms windows. We found that in our prediction, on average during a single eating episode, there were 337 transitions from *non-eating* to *eating*. As this is much higher than the ground-truth (average of only 18-19 gestures), we needed a second window to smooth out the noise from this classifier.

4.4.1.2 Determining Length of One Eating Gesture

From the ground truth data we found that on average an eating gesture lasted for 3.1 seconds (Rice - 2.8 sec, Noodles - 3.7 sec, Sandwich 3.1 sec) where a gesture starts from the point the hand starts moving upwards and ends when the hand comes back to rest. To determine if a gesture determined by the 500 millisecond window was actually eating, we take a window(w) of past raw classifier outputs (obtained every 100ms) and compare the number of *eating* gestures identified by the classifier during this window with a threshold (t) value. If the total number of classifications in w is more than t, then we declare the window to be an eating gesture window. We varied the length of the window (w) between 1 second and 10 seconds while the threshold was varied between 10 and 50 in multiples of 10. For example, for $w = 5$, there would be 50 classifications performed at the low level classifier (which gives an output every 100ms due to the 80% overlap). Table 4.6 shows the average error in determining the number of gestures (transitions from *notEating* to *eating*) in an episode, as a function of w and t. We computed $PredictionAccuracy = ((\Sigma GT - \Sigma P)/\Sigma GT) * 100$, where GT is the total number of eating gestures (ground truth) and P is the system-predicted gesture count. (A +ve value indicates that our system is under estimating, while a negative value indicates over-estimation.) From this table, we see the lowest values of error in gesture estimation are obtained for $w = 5$. A smaller value ($w = 2$sec) over-estimates the number of eating gestures, whereas an overly large window ($w = 10$sec) undercounts the number of eating gestures as it stays in the *eating* state for too long.

To understand if the optimal values for w and t varied based on food type. Table

w	t (threshold count)				
(s)	10	20	30	40	50
2	-152.1				
5	-4.2	-22.2	-3.4		
10	48.3	35.7	34.3	35.9	33.9

Table 4.6: Gesture Prediction Error (%) vs. (Window Size, Threshold)

		t (threshold count)				
		10	20	30	40	50
Rice	$w = 2$	-103.4				
	$w = 5$	0.36	-7.9	9.2		
	$w = 10$	63.3	39.3	35.3	47.1	42.5
Noodles	$w = 2$	-347.2				
	$w = 5$	-7.6	-84.5	-114.8		
	$w = 10$	53.5	38.9	25.1	-2.73	-19.1

Table 4.7: Differences in Gesture Prediction Error (%) Between Rice & Noodles

4.7 indicates the accuracy of determination for two different food types: Rice and Noodles. From the table we see that the estimation errors for different settings of w and t are indeed different, due to the different eating styles. (In case of noodles, the user usually holds the hand near the mouth till she has consumed the entire strand of noodle.) However, even though t and w varied across different food items, the variation was modest enough to allow us to use $t = 10$ and $w = 5$ seconds across food-types (i.e., for our gesture recognizer to be *food independent*).

4.4.1.3 Determining Eating Period

The next challenge was to determine the number of eating gestures that had to be observed in a fixed time period to declare that an eating episode was in progress. From the study, we had found that on average during a rice eating episode, an eating gesture occurred every ≈17 seconds. From the ground truth observation, we also saw that these gestures were not evenly distributed, but were rather bursty. Since we had to capture images of the food plate when we determined eating, we had to do it as early in the episode as possible. Our studies showed that, on average, the first minute of the rice eating episode had ≈ 3 eating gestures. Hence, we decided to

	Min	Max	Average
Rice	19	76	35
Noodles	13	70	31.2
Fruits	3	20	11.6
Sandwich	5	34	10

Table 4.8: Sensor Data Based Gesture Count Determination

detect an *eating episode* only if our system detected at least 2 eating gestures within the first minute.

4.4.1.4 Refining the Classifier

Building a Cost-Sensitive Classifier: When the base classifier (described above) was applied in User Study 1, it resulted in a high positive rate (see Table 4.3). This triggered detection of many false eating episodes and drained the battery rapidly by turning on the camera needlessly. To tackle this problem, we then increased the cost of false-positive misclassification in the training phase, thereby building a cost-sensitive classifier. However, as shown in Table 4.3, we now suffered from unacceptably high false-negatives (missing several real eating episodes). **Step2– Cost-Sensitive, Two-stage Classifier:** The following improvements were needed for version 3:

- We needed to determine the optimum cost for the classifier that provides the best trade-off between false positives and true negatives.
- We also needed an additional pre-classifier, that works on large frame size, to reduce the false-positive rate.

To get the optimum cost parameter, we first built the various J48 classification models for 5 values of cost settings $(0, 20, 35, 50, 100)$. Then we acquired day-long regular life-style sensor traces of non-eating activities from 3 participants (The participants were asked to remove their watches when they are eating and wear them at other times.). For the models with different cost parameter settings, the

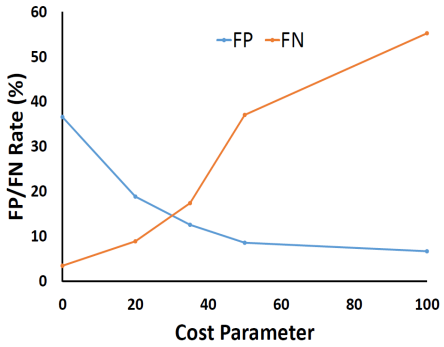


Figure 4.14: Error Rates for Different Cost Parameters

false negative rate ($FN/(FN+TP)$), was determined from cross-validation on the micro-study training dataset itself. To evaluate the false-positive rate ($FP/(FP+TN)$), we used the day long traces of non-eating data (from these 3 participants). Figure 4.14 provides the false-positive and false-negative rates for different values of cost parameter. When there is no cost, the FN rate is low, meaning we will not miss much of the eating gestures. However, the FP rate on real-life trace is very high (36.8%). For a cost of 100, the FP rate on the real-life trace is very low (6.7%), but the FN rate for eating is also very high (55.25%), implying we will miss most of the eating gestures. From this figure, we observed that a cost parameter of 35 provides a low value for both FP rate (12.61%) on the real-life trace and the FN rate (17.42%) for detecting eating gestures.

In addition, we observed that a large portion of the false-positives were generated by "jerky movements" of the hand during regular activities such as gesticulating during interactions, talking or repeated lifting of objects etc. While the image processing layer (described in Section 4.4.2) is able to filter out major portion of these false-positives, it still consumes significant resources on the smartwatch by unnecessarily triggering the camera. While a small frame-duration of $500ms$ is needed for efficient, low-latency triggering of the camera, an additional larger-frame duration of $2.5seconds$ was also needed to eliminate these other transient, short-lived gestures. Accordingly, we developed an additional classifier (Level 1, as shown in Figure 4.11) that uses a longer 2.5sec second frame of accelerometer data alone,

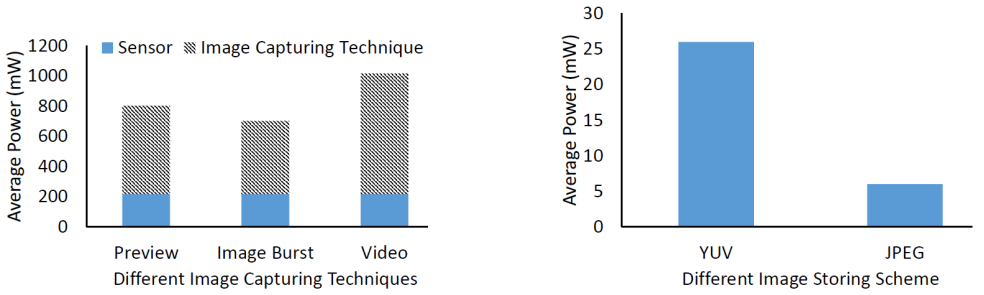
to first identify the *likely* eating episodes. To provide further robustness, the sensor readings are dampened via low pass filtering. As each eating episode is long-lived, this initial classifier can be used as a trigger for the fine-grained classifier (Level 2 in Figure 4.11) which works on the shorter 500sec frames, additionally using the gyroscope readings also. Once the eating gesture is consistently detected in level 1 (for more than 10 frames within a minute), this triggers the cost-based classifier (described earlier) that operates on $500ms$ frames. As shown in Table 4.3, the application of this two-stage classifier helped us to simultaneously reduce both the false-positive and false-negative rates.

4.4.2 Capturing Food Images

Via the feasibility micro-studies, we first focused on determining the best *mode* for capturing food-related images. Clearly, continuous video recording was not an option given the limited battery capacity of smartwatches (the Galaxy Gear 1 has battery capacity that is only $\approx \frac{1}{9}^{th}$ that of the comparable Samsung S5 smartphone): for continuous video capture, the battery drained out (from 100% to 10%) in ≈ 80 minutes. An alternative was to capture a single image–this however had two issues: (i) latency - the latency to trigger the camera and capture a single image was close to 900 msec, and (ii) precision - as the number of images captured was lower, the possibility of capturing an usable image (of the food) was extremely low.

We investigated the possibility of capturing *Preview frames*. Android exposes APIs which allows developers to grab the preview displayed on the screen. This preview refreshed at a high rate (more than 20 fps in the Galaxy Gear watch), thus solving the latency issue that we had with single image (or a burst of images). While of lower quality than that of a single image, we found the quality of Preview frames to be good enough for subsequent image analysis.

We also investigated the power consumption profile of these different modes. Figure 4.15 shows the power consumption (measured using the Monsoon Power



(a) Power Consumption: IMU (Sensor) + Image Capturing

(b) Power Consumption: Image Storage Scheme

Figure 4.15: Power Consumption by Various Components

monitoring tool [92]) for different approaches as compared to the baseline. The sensing energy is included for the image capturing measurements (Figure 4.15a)). From the figure, we see that the *Burst* mode consumes the least power, while the *Preview* mode consumed only marginally higher power. However, our feasibility studies showed that the burst mode could only capture an average of 2.7 images per gesture, while the preview mode captured 45.3 images (compared to continuous video, which captured 46.8 images). Given our desire for low latency, low power consumption and large number of captured images, we decided on the *Preview* mode as the most suitable approach.

We also studied the implications of storing the images on the smartwatch (the prior studies did not perform any storage). While the preview frames were in the YUV file format (approx. $150KB$), the files could be stored either directly in the YUV or in JPEG format (after conversion). Figure 4.15a) shows the energy consumption for the two techniques: JPEG not only resulted in significant power savings of around 80%, but also resulted in smaller file sizes (approx. $7kB$).

4.4.2.1 Image Filtering: Server-based

Our studies showed that many of the images captured by the preview mode were not *useful*–these included (i) blank images - when the camera captured only the table (ii) blurry images - when the hand was moving (iii) no food plate was visible and (iv) when neighbor's food plate or images with human faces was captured. We now

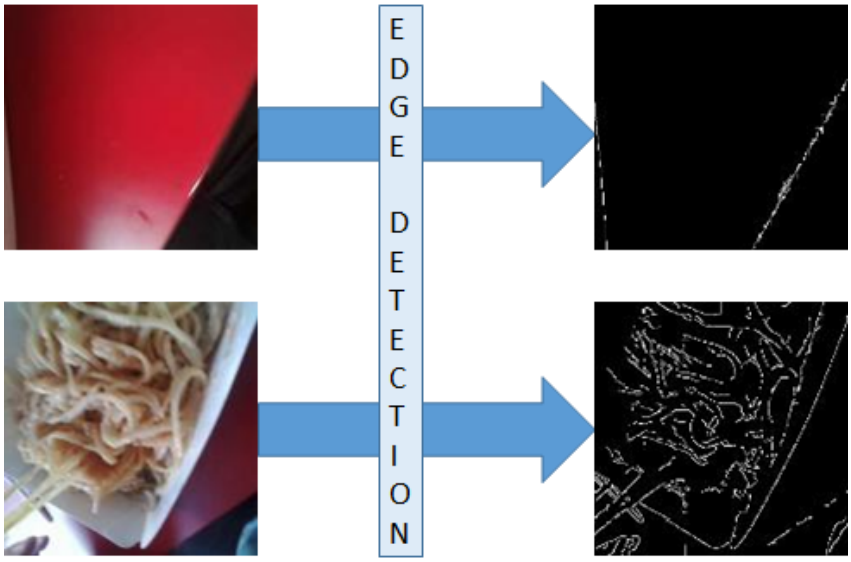


Figure 4.16: Output of Edge Detection

explain the sequence of steps (Figure 4.18 provides the pseudocode) that we used (at the *Annapurna* server) for filtering out irrelevant images.

Edge detection The first basic step applied is edge detection. When a clear image is captured, the image should have a large number of edges, where an edge can be the distinction line between the food items on the plate or even the distinction of the food plate from the table. We observed the following 'edge-related properties' for common cases of irrelevant images: (i) If the image captured is that of a solid background (the wall or the table), then the number of edges will be small; (ii) The number of edges in blurry images is smaller, compared to stable images. Based on these observations, the first step is to *eliminate images where the number of edges is smaller than a threshold.* Figure 4.16 presents two images as exemplars, where in the upper image, since the number of edges is less, it is discarded. Since the number of edges present in the lower image is more, it is retained for further processing.

Determine shape of edge The next step is to identify if the edge is the edge of a plate. Our assumption is that the plate has a regular shape (either rectangular or circular). To determine rectangular shape, we try to identify straight lines. When the number of pixels in a straight line (maximum deviation of ± 3 pixels from the ideal slope) is above a threshold, the associate edge becomes a candidate for a rectangular

(a) Originally Determined Bounding Rectangle

(b) Extrapolated Bounding Rectangle Shown in Blue

Figure 4.17: Bounding Box Extrapolation to Determine Maximum Area

plate. Similarly when there are a group of connected edges where the two end points are above a threshold arc length, then that edge is considered for curved edge. We use the $approxPolyDP$ function in $opencv$ to compute the number of curves in the edge. If the value is above a threshold, it is considered for a curved edge. We finally compute the slope values for consecutive edges, to determine if the shape is a regular curve: if the slopes do not exhibit a monotonic increase or decrease, that edge is no longer a candidate for the plate's outline.

Determine bounding rectangle and area For every regular shape that has been determined as a rectangle, the bounding rectangle is drawn around the shape (see examples in Figure 4.17)–in some cases, constructing this rectangle requires appropriate extrapolation of the edges. If the resulting extrapolated area is below a threshold, it is discarded. Similarly, extrapolation is performed for curved edges. For curved edges, the extrapolation will happen for two bounding rectangle corners and the extrapolation will touch either one or two edges of the image.

Eliminating non-food images and neighbor's food images using a depth map and CNN Several images were observed to contain edges, but from objects (e.g., pictures on the wall, or from the neighbor's plate) that were distinct from the user's food container. To eliminate such images, a depth map is constructed (via the parallax method) from the acquired sequence of images. First, two images that were

```
ALGORITHM 1: IMAGE RANKING
Input: Set of images from the phone: I(t)
Output: Best 10 images: BestImages
if (Currenttime − LastUploadTime < 900seconds) then
    I(t) = I(t) ∪ I(t − 1)
end
for Every image i in I(t) do
    if NoOfEdges ≥ 500 then
        NonBlurImages.add(i)
    end
end
for every image i in NonBlurImages do
    if image i does not have plate then
        NonBlurImages.delete(i)
    else
        if Food-plate in foreground then
            area = i.getPlateArea() i.setRank(area)
        end
    end
end
for every image i in NonBlurImages do
    if i.getRank() ≤ 10 then
        BestImages.add(i)
    end
end
```

Figure 4.18: Algorithm for Ranking Images

acquired 300 ms apart are taken. The dominant features in these two images are identified using SURF algorithm. Then these two images are rectified such that they are aligned along one of the axes. Now the pixel disparity between the features identified from SURF are evaluated to build the depth map: foreground objects have higher disparity than the objects in the background. If the rectangular/circular object detected in the image is in the foreground, then this image is saved as a likely image of the food plate; else, it is discarded. To further ensure that the image is indeed that of a food item, we then invoke the API provided by Clarifai inc. [27]. This API uses convolutional neural networks to identify the presence of food in an image.

Finally, all images that pass these filtering steps are stored, and ranked based on a 'visibility area' score: this score is directly proportional to the area of extrapolated rectangle, with an image with a larger score getting a higher rank. The average number of images being eliminated through each step of this filtering pipeline is provided in Table 4.9.

Filtering Step (921617 Images)	Device	Images remaining (%)
Total images captured	Watch	100
RGB Variance and Face count Filter	Phone	88
EdgeCount Filter	Server	37
Plate Shape Filter	Server	6.6
DepthMap and CNN based Filter	Server	0.8

Table 4.9: Effectiveness of Image Filtering

Figure 4.19: Images with Human Faces Detected

4.4.2.2 Lightweight Pre-processing on the Phone

While the above filtering algorithm can be effectively run on the server, it is too complex to be executed on the smartphone. In the absence of any pre-filtering on the phone, *all the preview images captured* would be transmitted from the phone to the *Annapurna* server. As this would unduly waste bandwidth (especially if the phone was not on a Wi-Fi network), we eventually (by User Study 3) implemented an additional lightweight pre-processing step on the phone. The pre-processor utilizes (a) a solid background detector, which computes the variance across pixels of the image, followed by (b) an initial face detection system using android's FaceDetector class. Images which had solid background or any visible human face were discarded. Figure 4.19 presents two images which were identified by the smartphone as images which contained human faces. When we ran this algorithm on the micro study images, we found that we could eliminate $\approx$12% of the images, even prior to transmission.

4.5 *Annapurna* Application

The *Annapurna* application consists of three modules: one each on the smartwatch, phone and the backend server. We first briefly describe our implementation of the three modules. We then also present the overall user feedback about using *Annapurna* across the three in-the-wild-deployments.

4.5.1 Watch and Phone Modules

The watch module is responsible for performing continuous gesture recognition and appropriately activating the camera to preview images. It is implemented to run on an Android smartwatch running Android version 4.3 or higher. There were several challenges in designing and building the watch module: (i) performing real time gesture recognition on a resource constrained smartwatch, (ii) On-the-fly bringing application to foreground to capture images (Android security requirement) and (iii) Ensuring all sensors were turned off when not in use were some of the challenges.

The parameters for all users were tuned to the default setting (Sec 4.3) of ($t = 10$, $w = 5$). In our current studies, we did not build a per-person classification model; instead, a single classifier was trained and deployed to all participants. The watch module also had a button to stop recording. This answered the user's privacy concern.

The phone component was principally involved in relaying the captured images back to the *Annapurna* server. The smartphone component was configured to perform batch transfer of images, and to re-initiate any interrupted transfers due to loss of connectivity. However, as mentioned previously, for User Study 3, the phone also included an image pre-processing engine that performed background and face-detection based elimination of images. This preprocessor was found to consume ≈ 0.37Joules per image, and incur a processing time of 267 msecs.

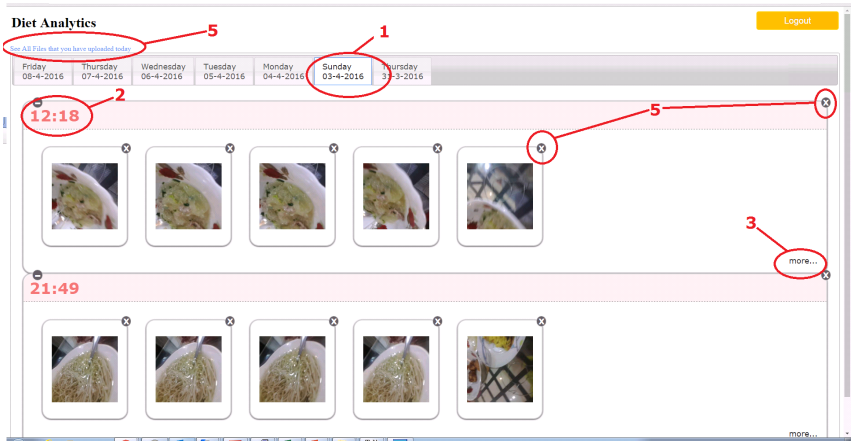


Figure 4.20: Snapshot of *Annapurna* Portal Shown to a User

4.5.2 Server Module and Parameter Choices

The server processes images sent by users to identify images of food and the *best* images determined by the server are stored and were later shown to the user in the food journal application. The users could decide if they want to accept the image or discard it. Figure 4.20 shows the food journal that was shown to a user once she had successfully logged into the system. In the web page, the user could navigate through tabs to get details of food consumed on a particular day (Label 1). In a particular tab, Label 2 shows all the meals consumed based on time for any day. Other than meal time, the user is presented with eating speed details as well as the number of spoons consumed during the meal. The user can expand the link indicated by *label 3* to view these details. Sometimes *Annapurna* predicts the wrong time as food time or the images that we show might not be a food image. The user can cross off a particular meal or a particular image by pressing the X indicated by *label 4*. Finally before we view any of the images, we allow the user to view all the images they have uploaded during a day. On clicking the link indicated by *label 5*, the user can view all files she has uploaded. These files are arranged date wise. If the user finds any inappropriate image, she can delete the image directly using this link.

The design of the *Annapurna* portal involved another question: *How many images per eating episode should be shown to a user?* To understand this, we sent

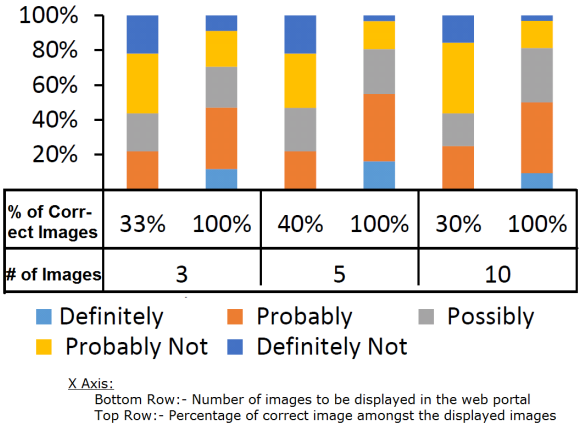


Figure 4.21: Number of Images to be Displayed in *Annapurna* Web Portal.

out a survey to the students and researchers on campus. We received 32 responses (16 males, 16 females). Figure 4.21 shows user response for different questions. From the survey we found that users prefer seeing fewer but accurate images rather than more (but potentially incorrect) images; accordingly, the *Annapurna* portal was configured to show only 5 images per eating episode to the user.

4.5.3 User Feedback and Opinions

Finally, at the end of the third *in-the-wild* study, we asked the participants questions regarding the usability and usefulness of the system. A total of 6 *Annapurna* users responded back; Table 4.10 tabulates their responses. From the response we found that most users found the overall system usable, but some users felt that the web application wasn't very user friendly. Regarding the number of images shown, most users agreed with the initial survey that 5 images were adequate.

When we asked participants to compare version 3 of *Annapurna* with the previous versions, all participants of *in-the-wild* study 3 felt that the system had improved since the previous versions in terms of energy drain of smartwatch. A major reason for this was the reduction in the duration for which the gyroscope was turned on. In the third user study, the gyroscope was turned on for an average of 136 minutes in a day for each user as compared to remaining 'on' continuously.

Users also provided the following feedback about the overall system, as well as

Question	Options	Response	No. of Users
The system was easy to use	1 (Strongly Disagree)– 7 (Strongly Agree)	6 7	3 users 3 users
Web Portal was self explanatory	1 (Strongly Disagree)– 7 (Strongly Agree)	5 6 7	2 users 2 users 2 users
Was it okay to show 5 images per meal	Yes No, more images should have been shown No, less images should have been shown	Yes More Needed Less Needed	4 users 1 user 1 user

Table 4.10: User Feedback for the Overall *Annapurna* System

suggestions for capabilities that they would like in future versions: (a) many users wanted a mechanism to automatically determine the calories that was consumed, (b) one user suggested that there should be a provision of having the watch app run without the phone app and (c) one user did not want the upload of the images to the server to happen automatically. Rather, he wanted to ensure that he inspected the images captured before they were sent. Out of these, *Annapurna* can be modified to support all the objectives except (a).

4.6 Discussion

There are certain open questions and challenges pertaining to automated food journaling applications. These includes:

Dominant Hand: One concern or limitation of the *Annapurna* application is that it currently requires the user to wear the smartwatch on his or her dominant hand. Anecdotally, there appears to be a reasonably significant group of users who prefer wearing the watch on their dominant hand. To study this issue further, we conducted a survey and based on information from 30 respondents we found that 67% of the respondents wore a watch and 50% of the watch wearers wore the watch on the dominant hand. Moreover, this assumption has been taken by various other researchers too (e.g. – [104, 145]).

Again, it is not necessary that the device worn has to be a smartwatch. Over the past few years, fitness bands (e.g. FitBit [36]) are becoming increasingly popular and with the greater consciousness towards health, the interest is bound to increase.

In future, if these devices are equipped with a camera, they can be used for the food journaling too.

Food Types Captured: In the current approach we have focused only on main meals, which are consumed in a plate. However, there are various other food habits which exist and *Annapurna* fails to capture images for those items – e.g. eating an ice cream. We have also noted that in our experiments with sandwiches, we found that even though based on repetitive hand to mouth gesture we could identify eating, we could not capture images in one-third of the cases. In such a scenario, it might be interesting to involve the user – e.g. say we cannot capture a useful image even after x gestures, we nudge the user to manually capture the image of the food item consumed. Alternately, we can just put a note for the user indicating that an eating episode was detected at a particular time instance. A similar technique can be applied for cases when a user wears a watch without a camera.

Similarly, we will miss capturing the image of the food plate when the hand gesture while consuming the food item is not repetitive or if there are long pauses between successive eating gestures.

Battery Life: For our current approach, the battery life is between 8 to 12 hours - depending on the number of times gyroscope and camera got turned on. My initial focus was to build the end to end system and to gradually optimise it. With the current battery life, we found that we could not cover all the meals because the battery drained out before the end of the day. Other than techniques described in the literature, I believe there can be system level tweaks which can improve the battery life. Some possible engineering tweaks can be - (a) Human behavior is usually routine and meals are usually consumed at certain specific places. If location based triggers (e.g. duty cycled BLE scan - maybe once in 15 seconds for known food location) can be used to turn on inertial sensor for eating detection, energy can be saved, (b) Smart duty cycling of the accelerometer - if it is detected that a person is sitting in a meeting, it is highly unlikely that the person will consume a meal (similar idea as ACE [97]) . So, other than the eating model, if the watch also runs alternate

activity recognition models and those models detect an activity which reduces the chances of a meal, then all inertial sensors in the watch can be put to sleep for a certain amount of time, based on the detected activity, (c) Currently we transfer all captured images from watch to phone. However, instead of transferring all images, if we transfer a subset of the images and based on the image, the phone can request for more images - e.g. if we send every second captured image (say image captured at time t and $t + 2$) to the phone and if the phone detects plate in two images, then it can request the watch to send the image that was captured between the two images (at $t + 1$) and the watch sends that image, then image transfer overload can be reduced.

Reducing Image Transfer: The existing image processing pipeline in *Annapurna* transfers every image captured by the smartwatch to the smartphone. The smartphone performs simple image processing to filter images which are either blurry or contain human faces. Images which pass through the filter are transferred to the server for robust image processing. Based on empirical analysis, we found that the smartphone could filter $\approx 12\%$ of the captured images.

A limitation of the existing approach is that there is an energy cost involved in transferring all the images from the smartwatch to the smartphone and transferring $\approx 88\%$ of the images from the smartphone to the server. Several approaches can be considered to reduce the overall image transferring cost. Some possible approaches include: (a) Moving parts of the image processing from the server to the smartphone. If the smartphone can handle some of the more complex processing steps that is currently done on the server, then the cost of transferring the images to the server can be mitigated. However, rigorous analysis has to be performed to understand the energy cost involved in transferring images to the server versus processing images on the smartphone, (b) The smartphone can transfer only a subset of the filtered images to the server. If the server can identify n*–good* images amongst the transferred images, then the server can notify the phone to terminate the transfer of the remaining images. Otherwise, the phone can transfer another subset of the remai-

ning images and this continues till the server identifies the n–*good* images or (c) The smartwatch can transfer only a few sample images to the smartphone. Based on the images received, the smartphone can determine the probability of capturing an image with the food plate amongst the previous or subsequent k images. If the probability is above a threshold, then the watch can send the subsequent images to the phone.

Additional Factors in Image Ranking: *Annapurna's* image ranking algorithm currently utilises variables such as number of edges in an image and the area of the plate to determine the best images. However, there can be several other variables which can assist in improving the image ranking algorithm. One such variable that can be utilised is the probability of the type of food in an image. Currently, in addition to whether an image contains food items, the Clarifai API also returns a confidence score for the type of food item in the image. The existing implementation of the image ranking algorithm utilises the value returned by Clarifai to assert if the image has food items. However, in future, the probability of the prediction by Clarifai can be utilised to determine the food item present in the image. This probability can also be an independent variable in the image ranking algorithm.

Personalization: Currently we have built models for a general group of people who have similar lifestyle. However, since a smartwatch is a personal device, a personal model deployed should improve the gesture detection accuracy and thus improving the overall system's accuracy. Since generally creation of personal models is more tedious, we can use a continuous learning technique, where we initially start with a general model, but gradually train the model with some correct eating gestures to improve the performance of the system.

4.7 Summary

In this chapter, I describe *Annapurna*, a system that we have developed for automated food journaling. For the automatic creation of the journal, we use a smartwatch's

inertial sensors for gesture recognition. Once an eating gesture is identified, the camera of the watch is turned on opportunistically to capture images of food. The captured images are processed to identify the best images which is finally presented to the end user. While developing the system, various system level challenges (e.g. handling sensor latency or improving energy) were addressed. Through *Annapurna*, we show that it is possible to build systems for automated food journaling (end-to-end ADL monitoring application) using off the shelf devices while addressing multiple system level challenges. As a next step, I plan to investigate techniques (similar to FoodAI [37]) to identify the food item in the captured images. The food item recognition will be helpful in providing a periodic summary of food items/types consumed by the individual as well as other similar analytics.

Chapter 5

Identifying Fine-Grained In-store Shopper Interactions

This chapter demonstrates the possibility of unobtrusively analysing the sensor data from multiple off-the-shelf devices to determine fine-grained context associated with the shopping activity. To identify these fine-grained contexts, we have developed the I^4S[1] system, which I will introduce in this chapter. The goal of I^4S is to identify objects (more precisely, the store shelf locations in the store) that a customer in a retail store interacts with, during a shopping episode. To realise this goal, I^4S utilises sensor data from diverse set of sensors embedded in multiple off-the-shelf devices – mobile, wearable and infrastructure. To address similar goals (identifying in-store interactions), existing approaches either use privacy-intruding techniques – e.g. monitoring the CCTV footage in the store [58] or other recording devices [122] or rely on manually surveying and observations [154]. We believe that the I^4S system will reduce the privacy concerns associated with customer's in-store activity monitoring, while ensuring fine-grained details of the activity can still be monitored. In this chapter, I primarily discuss the system design of I^4S and various challenges that we tackled while designing the system.

According to Applebaum [7], shopping can be described as a combination of

[1] pronounced I-foresee

Device	Model	Role (Best case Accuracy)	Sensors
Smartwatch	LG Urbane	Identify Picking Gesture (92.8%) Identify Shelf Level Location (89%) Identify Sub-Rack Level Location (92.4%)	Accelerometer Gyroscope Game Rotation Vector BLE scan
Smartphone	Samsung S V	Identifying Locomotion State (96.3%) Identifying Rack-Level Location (85.4%)	Accelerometer BLE scan
BLE Beacon	Estimote Beacons	Provide Location	BLE advertisement

Table 5.1: Devices Used in I^4S

two logically distinct activities: (i) inspecting or browsing items and (ii) eventually purchasing a subset of these items. Considering that I am interested in identifying fine grained context, I examine techniques to solve the first logical division of shopping - identifying items that a shopper is inspecting or browsing. To determine this fine-grained shopping context, I^4S takes a two step solution - (i) identify the "picking" gesture and (ii) identify the location from where the item was picked. Since a shopper can perform several gestures during shopping, I^4S has to ensure that picking gesture could be differentiated from the other similar gestures. In terms of location, several techniques have been proposed to determine indoor locations. Some of these techniques utilise the magnetometer. However, we found that the store where we conducted our user-studies had strong ferro-magnetic fields and thus we had to choose a technique which was less susceptible to such environments.

At a high level, the I^4S system utilises sensor data from the smartphone and smartwatch to determine the picking gesture, while sensor data from the smartphone, smartwatch and infrastructure sensor are fused together to determine the precise location (3-dimensional coordinates) where the picking occurred. Table 5.1 lists the devices used in I^4S along with their role and best case accuracy.

5.1 Necessity of Capturing In-Store Interactions

Before explaining the techniques to automatically identify a shopper item interaction (inspecting or browsing), let me reiterate the importance of identifying the interactions. Other than providing adequate information or feedback to the shopping individual, identifying in-store item interactions can provide various interes-

ting insights not only to the retailers, but also to sociologists who are interested in understanding shopping choices. For example, to answer the question: *"Does a shopper who visits a store to purchase an item which is being offered on discount also interact with other non-discounted items in the store?"*, Mulhern et al. [94] manually surveyed shoppers to found that there is indeed a positive correlation between interactions with discounted and non-discounted items when a person visits a store to purchase a discounted item. However, as this work required manual surveying, it does not have an exhaustive set of all in-store interactions, thus precluding the determination of further insights such as *which items are highly correlated* or *which items are always picked, but never bought* etc. On the contrary, online retail platforms not only digitally capture a user's click stream, but the entire browsing history, including time spent on different pages, navigation trend etc., and uses such history to enhance the platform's interaction with the user (e.g., personalized recommendations). To ensure that a physical retail store can offer a similar level of personalized, analytics-driven interaction to a shopper as an online store, there is a growing interest in using novel sensing technologies to capture a shopper's entire shopping behavior, including the item-level interactions that do not eventually translate into a purchasing act. Besides personalised analytics, identifying item-interactions will be an essential component of futuristic stores – e.g. Amazon Go [2], which provides a *checkout-free shopping experience*. In Amazon Go, shopper can enter a store, pick an item and walk out of the store, without bothering about standing in the checkout queue. To understand the possibility of identifying in-store interactions, in the rest of this chapter, I will describe our sensor-based approach which uses a combination of wearable, mobile and infrastructure sensors to identify the in-store item interactions.

Before proceeding further, we define three store-related terms that shall be used to explain how both the approaches work, and the type of interaction tracking that different components of the two approaches provides. These terms can be understood in the context of Figure 5.3, which displays an image of the commonplace

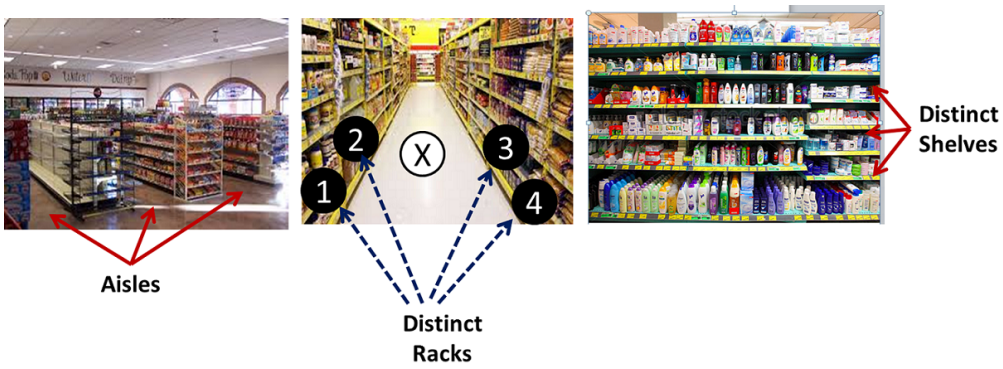


Figure 5.1: Store Layout and Distinct Terms

layout of items in a retail store:

- *Aisle:* A store typically consists of many rows of products, often arranged in a rectangular grid layout. An aisle is the passageway between such rows, through which shoppers navigate to inspect and retrieve items.

- *Rack:* A rack refers to a single modular unit of display (Figure 5.3 illustrates four different racks). An individual rack typically contains a variety of different products, although stores often organize items based on some logical grouping.

- *Shelf:* Shelf refers to a single level on a specific rack. Figure 5.3 shows an example of a rack with 7 distinct shelves. Shelves typically have a higher degree of product homogeneity – e.g., a particular shelf may stock only pasta, but possibly of different shapes and of different brands.

We next analyse the initial set up cost for any technology adoption that a shop owner will have to bear. To understand the price implication of various possible technologies to identify in-store interaction, we perform some back-of-the-envelope calculations. Figure 5.2 presents some in-store pictures from the stationary store in our campus where we conducted the user-studies. In the store, there are about 50 racks in the store and 6 shelves per rack, housing various stationary items – pens, notebooks, files, etc. Approximately 1000 pens are on display in each of the pen stands, while each of the note book racks had about 300 notebooks. We analyse the

Figure 5.2: Pictures to Estimate In-Store Item Density

cost implication of two technologies – a Bluetooth Low-Energy (BLE) deployment technology against a RFID based technology to identify the approximate cost that the shopkeeper has to incur for adopting any of the technologies.

Currently BLE beacons can cost anywhere between USD 5 to USD 50, with Estimote offering a wholesale price of approx. USD 20 for each beacon. In case we deployed a beacon in each shelf, the total cost for the entire store would be approx USD 6000, which is not a small investment. In case 1 beacon per rack deployment can solve the item interaction identification problem, then the set-up cost will come down to about USD 1000. Alternately, we could use RFID tags attached to each item, where a tag can cost anywhere between 5 to 15 cents. Assuming that a tag costs 10 cents and a pen stand will need about 1000 tags, tag deployment for a pen stand will cost approx. USD 100. Similarly, a 300 notebook holding stand will need tags worth approx USD 30. Assuming that on average, each shelf has about 500 to 750 items, the total cost for deployment will be approx USD 2500 to USD 3750. Other than the basic item cost, both these technologies requires a reader. Since the BLE approach requires a shopper's smartwatch/ smartphone for scanning/reading, no additional device cost is incurred by the store owner. However, in case of RFID, the store owner will have the additional overhead of purchasing RFID rea-

ders, where each reader can cost about USD 500 and depending on the deployment strategy (e.g. techniques similar to [135] require a dense deployment of readers (and the readers are more expensive – $\approx$ USD 1500) for every rack group), the number of required readers will vary. In case of fine-grained interaction understanding, the number of readers will be high. Thus deploying a RFID technology will cost more than a per shelf BLE deployment. Since our goal is to deploy the technology in-stores, where set up cost is a major factor, we go ahead with the BLE based infrastructure sensing strategy.

Shopping activity involves various interactions with in-store items. These interactions includes (i) picking an item from a shelf, (ii) putting that item in a shopping cart, (iii) returning that item back to the shelf, (iv) inspecting the item (e.g., reading the nutrition labels on a food product), or (v) evaluating the item (e.g., trying on a jacket to evaluate its fit). For this work, we focus exclusively on identifying "picks", as picking an item is the first concrete and strong example of shopper interest ([21] provides evidence that shoppers only pick up and interact with a small percentage (17%) of the items that they actually consciously browse in grocery stores).

I^4S involves innovative use of both the RF-sensing (of the advertisements broadcast by multiple beacons) capabilities of the smartphone and the inertial-sensing (using the accelerometer & gyroscope sensors) capabilities of the smartwatch. The smartwatch's inertial sensors are utilized to achieve two distinct objectives: (i) gesture recognition: identify the time instants when the user performs a "picking" gesture and (ii) fine-grained localization: determine the location of the user's hand at the instant when a picking gesture was performed. In this work I assume that if there is a separate backend repository that matches individual products with their on-shelf location, thereby identifying the item that a shopper interacted with.

(a) Shopper walking in the store (b) Stands in front of a Specific Rack
(c) Picks an Item from the Shelf and (d) I⁴S Functional Flow

Figure 5.3: Overview of I^4S System Working

5.2 System Overview

As mentioned previously, I^4S*'s* goal is to track all the item-related interactions that a shopper performs while visiting a store. To achieve the goal of identifying the products picked, I^4S must not only identify the individual 'picks', but must try to localize each such pick – i.e., it must resolve, at as fine a granularity as possible, the 3-D location in the store where the pick occurred. As a slightly relaxed, more practical expression of this ideal goal, I^4S should be able to identify at least the combination of (rack, shelf) where each pick occurred. I^4S makes the implicit assumption that there exists a database that contains the mapping between a product/item and the (rack, shelf) where it is located, and that there is thus a 1-to-1 mapping between a 3-D location coordinate and a product ID.

Given the twin goals of pick gesture identification and localization, we decided to devise the I^4S system based on a combination of infrastructure-mounted BLE beacons and a smartwatch mounted on the shopper's wrist. Our initial assumption was that (1) smartwatch scans of the BLE beacons would help provide accurate 3-D location; BLE was preferred over more traditional Wi-Fi localization as BLE typically has a shorter range and can thus be used for finer-resolution location tracking, and (2) the inertial sensors on the smartwatch would help us to identify the time instants when the shopper performed a 'pick' gesture. For various reasons (described in Section 5.3), this first-cut approach did not work.

The operation of I^4S eventually converged upon a more elaborate gesture-

triggered (rack, shelf) location tracking paradigm that additionally involved the user's smartphone (primarily to provide more robust BLE scanning capabilities and to provide an initial estimate of whether the shopper is sitting or standing). This paradigm consists of the following steps (Figure 5.3 pictorially illustrates the representative trajectory and actions of the shopper that correspond to these steps): (a) Shopper Moving: The shopper initially navigates through the store, moving around the various aisles. During this period, the I^4S application on the shopper's smartphone continues to collect the Received Signal Strength Indicator (RSSI) information of nearby beacons via BLE scanning, but does not actually attempt to localize the shopper; (b) *Shopper Stops at a Specific Rack:* Once the shopper has identified a specific product of interest, she stops in front of a specific rack. The shopper may continue to stand or may sit down, to look at products on the lower shelves. At this point, I^4S continues to remain in passive sensing mode. (c) *Shopper Picks out Item from a Specific Part of a Specific Shelf:* This corresponds to the "pick" activity instance that we seek to monitor. It is at this point that the context determination logic of I^4S (described below) is actively triggered; (d) *Shopper Resumes Browsing Activity:* After the pick instant, the shopper continues with the rest of her actions, which may involve continuing to remain stationary at that rack, or moving on to other parts of the store.

The current I^4S system determines the occurrence and 3-D location of the "pick" activity through an offline process, where after the completion of the shopping, the entire data trace is extracted from the devices of the shopper and we analyse the trace to determine the "picks". However, this is not a system limitation. In future, the entire logic to determine the picking gesture and picking location can be implemented in an individual's mobile device to track the picking actions in near-real time (within 5-10 seconds of the actual occurrence of the pick). The steps for determining each pick's 3-D location is as follows:

1. *Identify Pick Gesture:* I^4S first uses the stream of accelerometer and gy-

roscope data collected from the shopper's smartwatch to both infer the occurrence of a "pick" gesture, and the time that the shopper performed this gesture. To improve accuracy, such pick gestures are identified only when the user is stationary (as determined via inertial sensing on the shopper's smartphone).

2. *Localize to the Corresponding Rack:* I^4S then uses the recent history of BLE scan data (and potentially even the BLE scan for the next few seconds after the occurrence of the pick gesture), captured by the smartphone, to retrospectively compute the rack from which the pick occurred. Note that this determination directly identifies the rack, instead of a specific location coordinate: as we shall see later, an alternative method of determining the rack implicitly via estimating the shopper's orientation from magnetometer data is ineffective due to the significant ferromagnetic noise in stores.

3. *Localize to the Shelf Level:* After the rack has been identified, the I^4S App uses the accelerometer data (corresponding to the time when the pick occurred) of the smartwatch to determine the shelf level of the "pick". To improve the accuracy of such shelf-level classification, the smartphone data is used to create a prior of whether the shopper is sitting or standing. Note that the determination of the shelf level is done via a *classifier*, rather than the conventional method of using BLE-based localization, as BLE localization did not provide the required level of location accuracy.

4. *Localize Within the Shelf:* To further improve the 3-D localization of the "pick", I^4S subsequently tries to distinguish between various sectors of the same shelf. More specifically, in our experimental studies, we associated the left and right halves of a shelf with two distinct sectors, and then use the game rotation vector data (obtained from the smartwatch's gyroscope sensor) to classify the pick between these two sectors.

Figure 5.4 illustrates the flow of the system, including the various contexts sen-

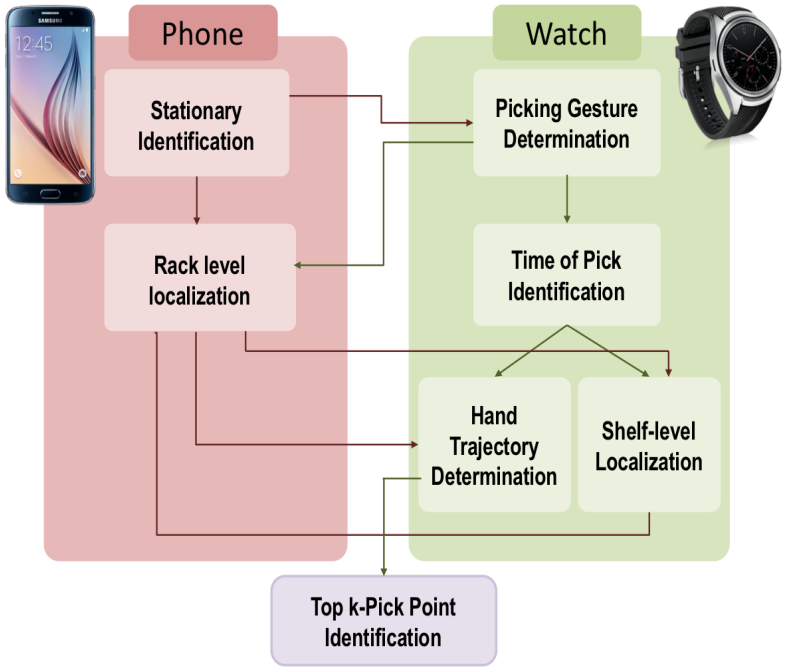


Figure 5.4: Overview of the I^4S System with Smartwatch and Smartphone

sed from the smartwatch and the smartphone. Note that most smartwatches currently operate by pairing with a smartphone. We can thus expect that the shopper will have both a smartphone and the smartwatch during a store visit.

5.3 Design Choices

To understand the feasibility of identifying shopper's interaction, two types of studies were performed: (a) *Lab Study*: an initial study was performed in our lab's pantry where lab members mimicked shopper's behaviour and (b) *In-Store Study*: student volunteers were recruited and incentivised to perform the study in a stationary store on the SMU campus. The image of the lab's pantry is shown in Figure 5.5, while the image of a part of the store is in Figure 5.6. Next, I will provide the details of the dataset and the initial findings which justified design choices taken for I^4S.

Figure 5.5: In-Lab Data Collection Location

Figure 5.6: Shop Where Data Collection was Performed

Parameter	Value
Number of participant in the study	31(14 males)
Number of shop visit data used	25
Total shop visit duration	2 hours 52 minutes
Total number of picks	778
Number of racks from where items were picked	43
Number of beacons deployed in store	35

Table 5.2: Summary of Dataset Collected In-Store

5.3.1 Dataset

Lab Study The intention of this study was to justify various design choices taken. For this study, members of the lab were recruited as participants. The participants were asked to wear a smartwatch (LG Urbane) on their dominant hand and carry a smartphone (Samsung Galaxy S6) in their pocket. Four BLE beacons were placed in the pantry to understand the feasibility of using the RF signals emitted by the beacons to identify a participant's smartwatch's and smartphone's location. Participants were asked to perform various directed ephemeral tasks. Some of these tasks included picking items from different shelves, picking one item multiple times from one shelf, picking multiple items after walking around the lab during successive picks, etc. . During the tasks, inertial sensor data (accelerometer, gyroscope, and magnetometer sensors) from both the smartwatch and the smartphone was collected.

In-store field study: For the in-store data collection, we recruited 31 students from our university through email invitations, with IRB approvals. The students who agreed to take part in the study were first briefed about the study, then given a smartwatch and a smartphone with our custom application running for data collection. They wore the watch on their dominant hand and carried the smartphone in the front pocket of the pants. There was no specific task that was assigned to the shoppers while they were in the store and they were free to walk around the shop without any time limitation. As compensation, we provide each participant a shopping voucher worth $10 which the shopper could redeem at the store. We termed each such visit to the store as an *episode*.

Before the data collection, we instrumented the shop with 35 Bluetooth Low-Energy (BLE) beacons. All beacons were placed on the ground level at the base of the racks. We set the beacons with a transmission interval of 101ms and a transmission power level of -20dBm.

For our analysis, we used sensor data from 25 out of the 31 shopping episodes. Four participants did not carry the smartphone (not wearing clothing with pocket) and two participants' data had data synchronization issues for the data collected in the smartphone and smartwatch and thus their data was omitted from our analysis. A total of 778 picks from 43 distinct racks were observed during this data collection. Table 5.2 summarizes the dataset that was collected from the store visit.

Ground Truth collection: The ground truth for this study was collected by shadowing the shopper. The shadower used an application running on a Samsung Galaxy Note Pro 12.2 LTE device to record the micro-activity labels of shoppers. The application provided buttons to mark "Standing", "Picking", "Bending", "Sitting", "picking from left", "picking from right". Other than this, the screen had a provision to mark the "rack location" and the "picking shelf". Figure 5.7 shows the screen for the application with labels overlaid for explanation. Label 1 Indicates the top view of the shop. In this view, location ground truth can be marked by touching position in the layout corresponding to the location of shopper in the shop. Label

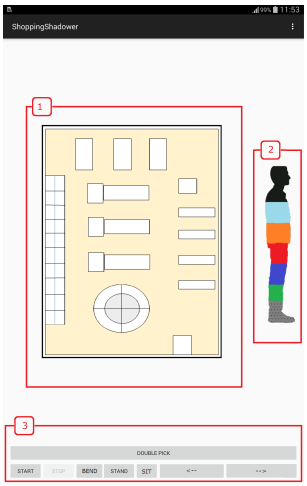


Figure 5.7: Ground Truth Data Collection Application Screenshot

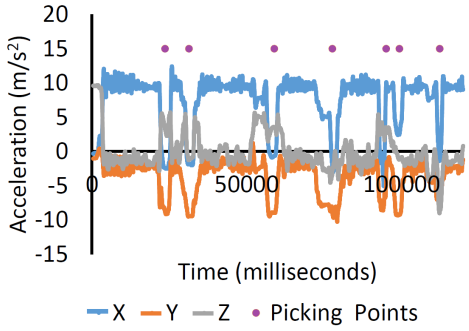


Figure 5.8: Smartwatch's Accelerometer's Data Variation for a Shopping Episode

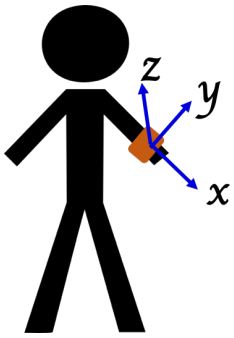


Figure 5.9: Orientation of Different Axis when the Watch is Worn on the Hand

2 allows the shadower to mark the position of the shelf from where the item was picked. The human figure is only for reference. In case a shopper picked an item from the lowest shelf, then the light grey dotted section in the figure is marked and so on. Label 3 consists of various buttons which can be clicked to mark the ground truth.

5.3.2 Inertial Sensor Analysis for Gesture Recognition

We inspected the accelerometer data collected from one participant who picked items from shelves in our lab's pantry. Figure 5.8 shows the variation smartwatch's accelerometer data along with the pick times when the person is conducting a series of picks. From the plots we can see that whenever a pick occurs, there is a large

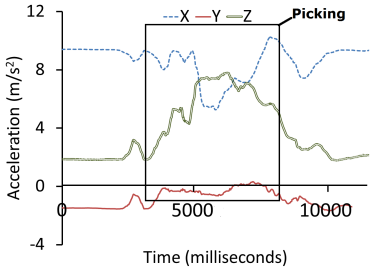


Figure 5.10: Accelerometer Variation for Picks from a Lower Shelf Under Controlled Conditions.

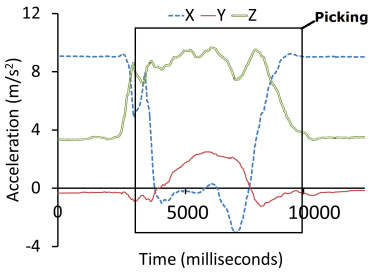


Figure 5.11: Accelerometer Variation for Picks from Top Shelf Under Controlled Conditions.

variation in multiple accelerometer axis indicating that it might be possible to identify the picks. We next looked at all the shopping episodes and found that a picking gesture (hand moving from the resting position to the item and coming back to the original position) lasts for ≈ 4 seconds on average. We thus used this as the window size for all feature extraction.

An interesting observation from the data inspection was that picking from different shelves resulted in distinct changes in each axis of the inertial sensor. We plotted the variation of accelerometer data when a person picked from the top shelf (approx. 1.5 m from ground level)(Figure 5.11), versus picking from a lower shelf(ground level)(Figure 5.10). From the figure we can see that there is visible accelerometer variation in the accelerometer data in both the cases. However, the changes in each axis is different during the picks from different shelves. This indicated that it might be possible to identify the shelf from where an item is picked.

5.3.3 Bluetooth Low Energy (BLE) Analysis

We next analyse the characteristics of the BLE beacons heard. Since both smartwatch and smartphone has the capability of scanning BLE beacons, we analyse the characteristics of beacons heard by each device to determine if we could use either or both the devices for localisation. We compare the two devices in terms of beacon listening capabilities - i.e. is there more packet loss, the RSSI of beacons heard, the

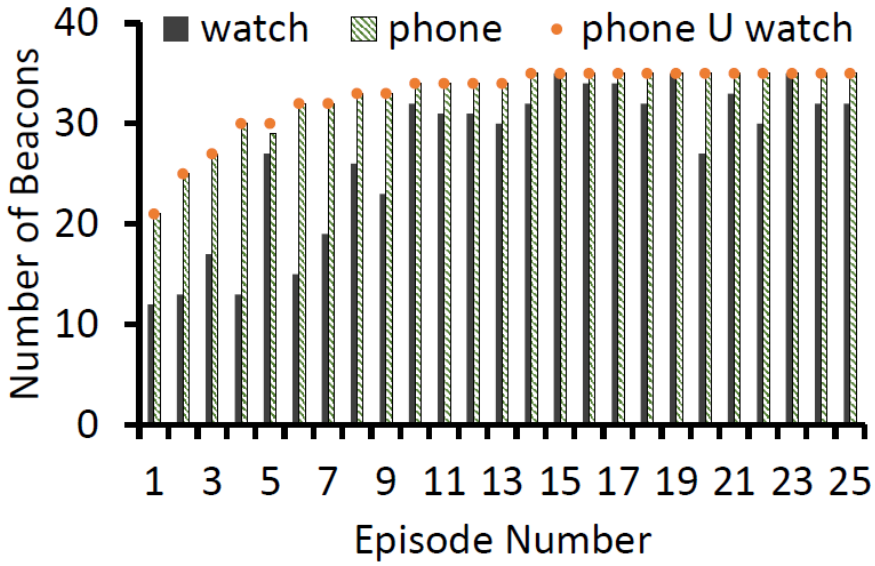


Figure 5.12: Difference in Number of Beacons Heard by Phone and Watch

duration during which the beacon is heard, etc.

We first analyse the miss rate for the two devices. For this analysis, we use BLE data from the 25 episodes of the in-store study. For every episode, difference in number of beacons heard can be subdivided into two categories: number of unique beacons heard in an episode and count of total beacons heard. Figure 5.12 shows the difference in number of beacons heard by the two devices in an episode. In the figure, the solid bar represents size of the set of beacons heard by the watch, while the striped bars represent the size of the beacons heard by phone. The dot for every episode represents the count of the union of set of beacons heard by the phone and set of beacons heard by the watch. From the plot, we find that for almost all episodes (except episode 5), the number of beacons heard by phone is equal to the union of beacons heard by phone and watch, indicating that if a watch hears a beacon, it is almost likely that the beacon will be heard by the phone. However, the opposite is not true as in certain episodes (e.g. episodes 4 and 6) not even half the beacons heard by the phone were heard by the watch. From the figure we can also see that only in ≈65% episodes, the phone hears all the deployed beacons.

We next analyse the difference in the scan results obtained by the two devices. For an entire episode we count how many distinct BLE identifiers (not unique)

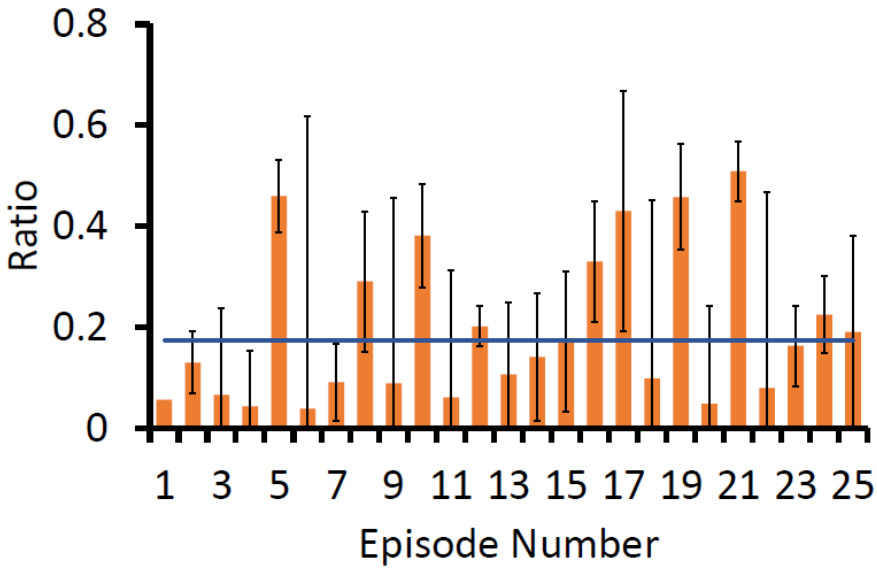


Figure 5.13: Episode-wise Ratio of Beacons Heard by Phone and Watch

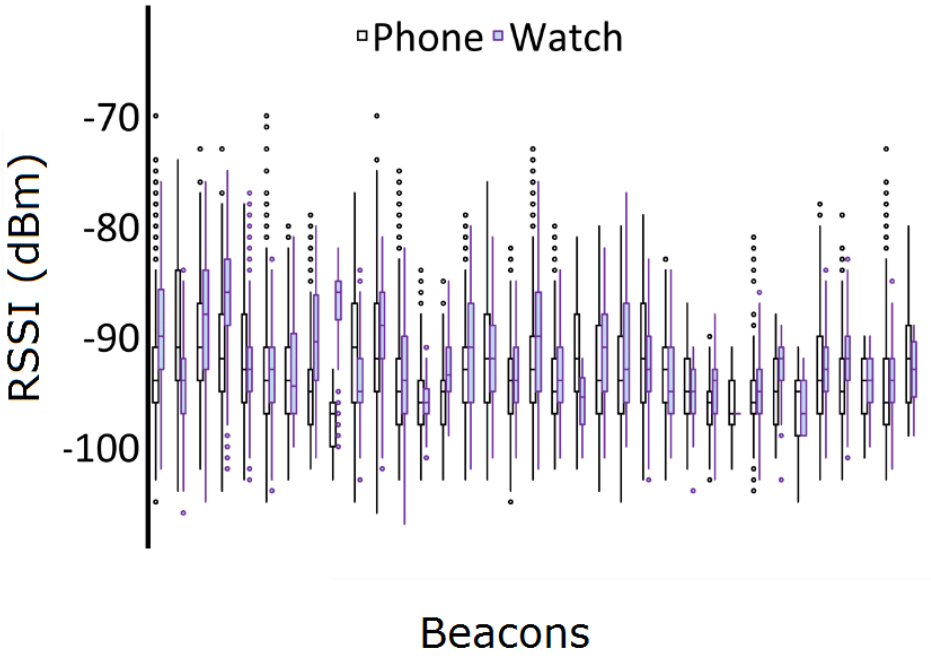
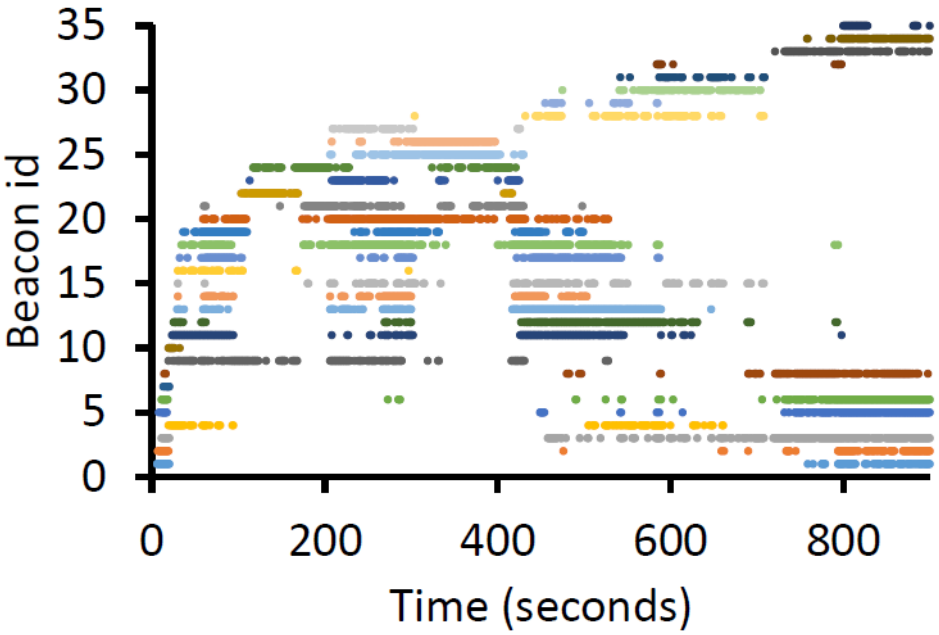
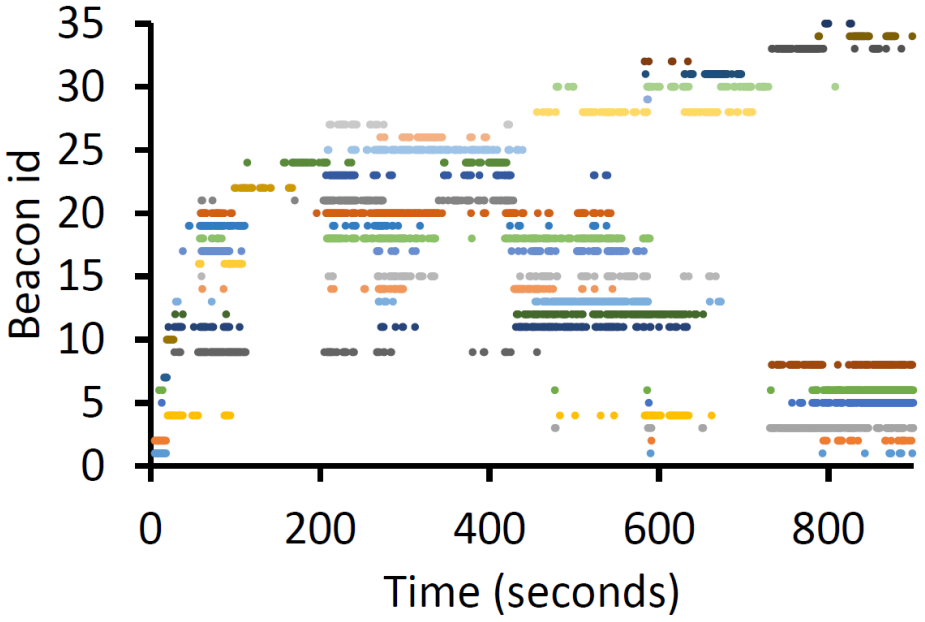


Figure 5.14: Difference in Received Signal Strength Indicator (RSSI) Between Phone and Watch for an Episode



(a) Phone



(b) Watch

Figure 5.15: Timeline Showing When a Beacon was Heard by the Device

were recorded by the devices. Figure 5.13 plots the ratio between the total number of records heard on phone and watch (with standard deviation). In the figure, the orange bars represents the average ratio for a particular episode (with error bars for standard deviation) and the blue line running across the figure represents the average of the ratio, which was equal to 0.173, indicating that the phone hears approximately 6 times more beacons than the watch.

We next compare the difference in RSSI for the beacons heard by the two devices. To understand this, we scrutinise data from one episode (episode no. 15), where all beacons are heard by both the watch and the phone and the ratio between the number of beacons heard is almost equal to the average of the ratio across all episodes. Figure 5.14 shows the box plot of received RSSI for episode 15. The black bordered box plot represents the data heard by the phone, while the purple box plot represents the data heard by the watch. From the plot we can see that there is actually not much difference in the inter-quartile range of signal strength heard by the two devices. However in terms of maximum and minimum RSSI heard by the devices, we see that the phone has a wider range for most beacons. Finally we plot the time-line to understand when different beacons were heard by the phone and watch. Figure 5.15 shows when different beacons were heard by the two devices. The X axis is the time series, while the Y axis represents a beacon. From the two sub-figures we can see that even though the overall periods when a beacon was heard is similar, however Figure 5.15b is more sparse than Figure 5.15a. This indicates that even though the smartwatch misses a lot of beacons, yet it is able to maintain the RSSI distribution similar to that of the smartphone.

From these studies, we see that a smartphone has a better beacon capturing capability, which convinced us to pursue the direction of using the smartphone to identify the location of a person.

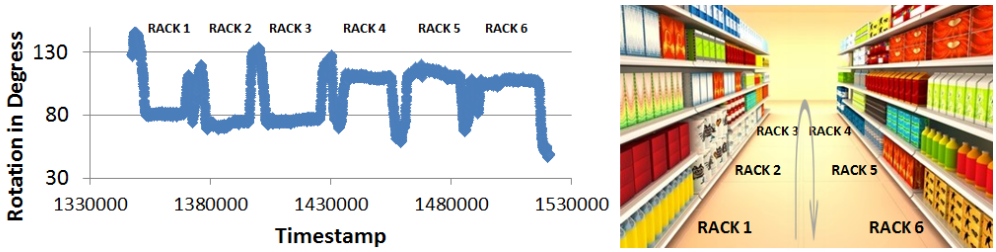


Figure 5.16: Variation of Magnetometer Readings Inside the Store

5.3.4 Magnetic Field Sensor Analysis

In case of a narrow aisle with racks in each side, a position sensor (compass) could help in identifying if a person was facing a particular rack or a rack that was 180^o opposite to it. We wanted to understand if this argument held true in case of the store where we collected data. From the initial study, we extracted compass data value from a store visit. During this visit, the person stood in front of various racks wearing the smartwatch on the wrist and the face of the watch (z axis) facing the rack, while holding the hand still. Figure 5.16 shows movement of the shopper in the store, where shopper's trajectory was : Rack 1, Rack 2, $\cdots$ Rack 6 and the variation of the compass for this trajectory (with the Rack level location indicated). From the plot we can see that when a person's orientation changed by 180^o, the compass had a maximum variation of 60^o indicating that strong ferro magnetic fields in the store affected the readings from the compass. We believe that we could have major location prediction inaccuracies, especially when racks are not diametrically opposite to each other, if we used the readings from the magnetic field sensors and thus decided not to explore the compass for our experiments.

5.4 Methodology

To identify in-store interactions, I^4S relies on inertial and BLE scan data from a smartwatch and smartphone. There are three main components to the entire system:

- Identifying the picking gestures.

Feature	# Distinct Features	Description
Mean	4	Average of the values of the axis data in the time window and the average of the magnitude
Variance	3	Variance in the values of the axis data in the time window
Mean crossing rate	3	Count of times the values cross the window's mean
Max mean	3	Divide the window into sub-windows and compute the maximum of the means of the sub windows
Max rise	3	Divide the window into sub-windows and compute the maximum positive change in mean in consecutive sub-windows
Max drop	3	Divide the window into sub-windows and compute the maximum negative change in mean in consecutive sub-windows
Covariance	3	Co-variance between the axes of the sensor
Entropy	3	The spectral entropy of the axis data in the time window

Table 5.3: Features Extracted from Inertial Sensors

- Identifying the rack in front of which the shopper was standing while picking an item.
- Identifying the shelf and the zone within the shelf from where the item was picked.

In this section we shall elaborate on the techniques used in solving the above-mentioned objectives.

5.4.1 Pick Gesture Detection

The first step in the pipeline is to identify the picking gesture. We use a standard activity recognition pipeline for this detection.

Preprocessing and Framing: The accelerometer and gyroscope data from the smartwatch and the smartphone is extracted. The accelerometer provides acceleration of the device for 3 perpendicular axes, while the gyroscope provides the speed of rotation about each of the axis. The data from both the devices are divided into frames of length w with 50% overlap between frames. Every instance of the frame is a tuple represented by [$time$, $accel_x$, $accel_y$, $accel_z$, $gyro_x$, $gyro_y$, $gyro_z$].

In order to identify the picking gesture, we used the sensor data from the smartwatch. For each frame we computed statistical features for each sensor axis in the frame as described in Table 5.3. Finally for each frame we had to identify the label. From empirical observations we identified that on average, the hand moved from a

position of rest to the item of interest and back to rest in about 4 seconds (with a range from 2 seconds to 10 seconds). This duration varied depending on the distance between the initial position of the hand from the object of interest, the time spent in inspecting the object before actually bringing it back to the trolley or leaving it in the shelf. For the frames extracted from the smartwatch's sensor data, time of picking is marked as the time when the hand touched the item of interest. The time of picking was extracted from the ground truth files for the episode. Frames which were within ± 2 seconds of the pick marked in the ground truth data were labeled as picking frames.

Similar data was extracted from the smartphone sensor. However, the label for the frame was one amongst $-$ {walking, standing, bending, sitting}

Gesture Recognition: After extracting frames from the shopping episode and labeling the frames, we used a classifier in identifying picking gesture from the smartwatch data. We used weka [41] for our classification. Various classifiers were tested for performance and we found that we could achieve reasonable accuracy in identifying picking using a Random Forest [45] classifier. We thus used the same in our studies.

While observing shoppers in a store, we found that most picks occurred when a person either completely stationary or had very small movement. From our data we identified only 4 instances of picks (out of the 778 total instances) where a shopper picked an item while moving past the shelf. This suggests that the pick gestures should be identified only when the user is relatively stationary; this additional context predicate on the shopper's locomotive state helps eliminate false positives generated due to random hand movements.

To identify locomotion, the data from the smartphone was used. We used the same classifier in predicting the shopper's locomotion state identification. While evaluating our system, we found that in certain situations the four locomotion states had to be sub-grouped. For example, we had observed that while picking an item, a shopper is usually stationary. In this situation, we re-labeled all *stand*, *bend* and

sit labels as *stationary* and re-ran our classifier. Result from this classification was used in conjunction with the smartwatch features to identify picking.

5.4.2 Rack Level Pick Location Identification

As mentioned previously, we had deployed Estimote's BLE beacons inside the store and the smartwatch as well as the smartphone continuously scanned for the beacons and locally logged all discovered beacons along with the corresponding RSSI. RSSI is a function of distance, with the RSSI being lower when a person is far from a beacon. Due to results shown in Section 5.3.3, as well as our analysis of the smartwatch's BLE scan data for 3-D location identification (using the approach mentioned in this section for 3-D location tracking for a granularity 100 cm x 50 cm x 30 cm, we found that using a smartwatch, we could identify the correct location in only $\approx$13% picks as compared to 2-D location accuracy of $\approx$ 60% picks identified by smartphone without using history based prediction techniques), we decided to pursue the smartphone based shopper's in-store location determination, rather than using the smartwatch to determine the shopper's hand's 3-D position when an item is picked. A limitation of using the smartphone is that the smartphone can only identify the shopper's physical location (rack of width 1 meter, in front of which the shopper is standing), but can not identify the hand's location (i.e. what item is picked). In the store where we performed our studies, most racks are 1 meter in width and the racks were arranged shoulder to shoulder. Each shelf within the rack is approximately 30 centimetres high. For a smartphone based location tracking, since we aim to identify the rack in front of which the shopper is standing, we compute the system's accuracy based on whether the system could identify the rack in front of which the shopper was standing, which is approx. 1 meter.

To identify the in-store location, we extracted frames of size w ($w/2$ seconds before and $w/2$ seconds after a pick was marked) from BLE scan log of the smartphone. Similar to gesture detection, we used a window of size $w = 4$ seconds in

our evaluation. For this window w, we computed each deployed beacon's average RSSI (35 values). If a beacon was not heard in the window, the RSSI value for the beacon was set to a very small number. Finally a label indicating the location (rack) from where the shopper is picking an item is added to the vector. The length of the vector (denoted as M) in our case was 36 (35 beacon information + 1 location information) and this vector represented a fingerprint of the location. This step was repeated for all P picks that took place in a shop (778 picks in our study) and we created a fingerprint map of size $\{M \times P\}$. The same location could be represented by multiple entries in the map.

To identify the location of the shopper we used a RADAR like approach [9] where, once the fingerprint map was generated, for each pick P_i, we computed the euclidean distance of the fingerprint P_i from all other $P - 1$ fingerprints in the fingerprint map. The fingerprint is assumed to be at the location which has the smallest euclidean distance. For evaluation, we noted down the top-k smallest distances for each fingerprint and then assigned a probabilistic location to each fingerprint instead of a deterministic one, where the probability was computed as the inverse of the euclidean distance between the testing fingerprint and the k^{th} closest fingerprint.

Since we had historical movement information of the shopper, we computed the M dimensional vector not only for the time window when the pick occurred, but also h windows of size w before the pick. For each of the landmark, we applied the RADAR like location identification algorithm to determine the person's position for that window. Due to physical constraint, a person can traverse a certain number of grids at max in a unit time. For example, if two grids are 5 m apart inside a shop, it is highly impossible for a person to traverse between these two landmarks in a few milliseconds. To realise this intuition in our algorithm, we performed a Viterbi smoothing [147] on our data. A trellis diagram (Figure 5.17) corresponding to the viterbi decoding was generated to determine the most likely sequence of locations that a shopper was assigned just before the pick occurred. For the viterbi implementation, we used a depth of ($h = 4$), indicating that we were using 4 time windows for

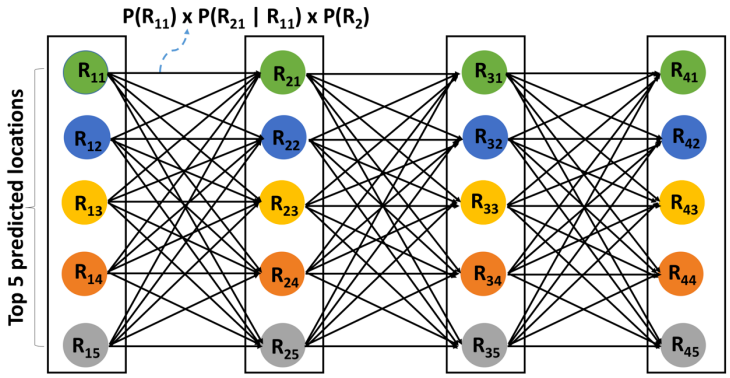


Figure 5.17: Trellis for Viterbi Smoothing

our prediction. For our trellis, the top-k prediction by the RADAR technique was used as the nodes for each level L, while the edges were computed as the product of the probability of the prediction for the note at the current level, the probability of the node at the previous level and the transition probability from node at previous level to the node at the current level. The transition probability used in our study was computed empirically from the ground truth file for each transition that took place during any of the shopping episodes. Alternately, similar to [53], we could consider using the inverse of the distance between racks as the transitional probability. Based on our current approach, the path with the highest probability was selected as the most likely path and the node at level h was identified as the correct location of the shopper.

5.4.3 Shelf Level Pick Location Identification

Identifying the part of the shelf where the pick took place can be divided into two sub-parts: (i) Identifying the shelf from where the picking took place and (ii) Identifying the location within the shelf from where the item was picked.

Identifying the Shelf from which the item was picked: In Section 5.4.1 we listed the steps involved in identified a picking gesture. In order to identify the shelf from which item was picked, we extracted all the frames which were labeled as picking. An additional field was added to the frame:- the shelf level from which item is

picked. Since the racks in the stores had varying height and varying number of shelves. For consistency, we labeled shelves as: L1 - if shelf was 0 to 30 cm from the ground, L2 - if shelf was 30 cm to 60 cm from the ground, $\cdots$ L6 - if shelf was 150 cm to 180 cm above the ground. In case of shelves which were more than 30 cm high, we labeled picks from it as L_{lower} if the shopper picked an item from the lower half of the shelf and L_{higher} otherwise. We passed these frames to the Random Forest classifier which identified the shelf from where the item was picked.

Identifying the pick zone within the Shelf: Finally, to identify the point from where a shopper picked an item, we plotted the trajectory of the hand while picking an item. Android provides a virtual sensor - Rotation Vector, which uses the data from the 9 axis IMU sensors (accelerometer, gyroscope and magnetometer) to provide the quaternion value which can be used to determine the hand's trajectory. However, since the magnetometer was highly influenced by the items in the store as well as the material used in the store, we decided to use the Game Rotation Vector sensor, which is identical to the rotation vector sensor, except that it does not use the geomagnetic field.

The output of the Game Rotation Vector is also a quaternion value. A quaternion represent the orientation and rotation of an object in a 3 dimensional space. Since the Game Rotation Vector does not use the magnetic field sensor, the output of the quaternion does not provide results with respect to the Earth's magnetic north. The Game Rotation Vector provides a quaternion value which gives us the axis and degree of rotation of the watch in a 3D space. A Game Rotation Vector(q) is a unit quaternion of the form:

$$q = cos(\theta/2) + x \cdot sin(\theta/2) \cdot \hat{i} + y \cdot sin(\theta/2) \cdot \hat{j} + z \cdot sin(\theta/2) \cdot \hat{k}$$

where (x, y, z) represent the axis of rotation and θ represents the angle of rotation. Since the game rotation vector needs an initial reference point, the hand is positioned at a fixed orientation and location at the starting of every shopping ses-

Feature	# Distinct Features	Description
Displacement	6	Displacement of the wrist in all the axis during first half of gesture and second half of gesture
Distance	2	Distance from the location of the pick from the starting and ending point
MeanVel	2	Mean velocity of the wrist during first half of gesture and second half of gesture
MaxVel	2	Maximum velocity of the wrist during first half of gesture and second half of gesture
MedianAng	6	Median of angular velocity for yaw, pitch and roll during the first half of gesture and second half of gesture
MaximumAng	6	Maximum of angular velocity for yaw, pitch and roll during the first half of gesture and second half of gesture
NetAng	6	Net angular change for yaw, pitch and roll during the first half of gesture and second half of gesture

Table 5.4: Features Extracted from Game Rotation Vector Sensor

sion. Any point that is derived further is with respect to this reference point. An advantage of using the game rotation vector in a magnetic environment is that the relative rotations provided are more accurate as compared to the rotation vector. For our studies, we used the unit quaternion given to us by the Game Rotation Vector Virtual Sensor to rotate this point in 3D space to get the final coordinates of the wrist. A point p(p_x, p_y, p_z) in 3D space can be rotated using a quaternion using the following formula:

$$p' = q \cdot p \cdot q^{-1}$$

where q^{-1} is the inverse of the quaternion q and can be expressed as:

$$q^{-1} = cos(\theta/2) - x \cdot sin(\theta/2) - y \cdot sin(\theta/2) - z \cdot sin(\theta/2)$$

Hence, if w_t represents the wrist position at time t and q_t represents the value given by the Game Rotation Vector Virtual Sensor, then w_t can be calculated as follows:

$$w_t = q_t \cdot w_0 \cdot q_t^{-1}$$

For a picking gesture, we extracted the quaternion values from the smartwatch data. Again a window of ± 2 seconds was used to extract the quaternions. For each of these 4 second window, we computed the position of the wrist at all times Δt in

this window. A spline was used to fit in all the predicted points in the trajectory. For each trajectory we extracted features as mentioned in Table 5.4. The features are similar to features used in [104], except that we do not use the duration features and in addition to the *roll* and *pitch* features, we also compute the *yaw* features. The yaw is useful in a shopping scenario as it can help in determining if the hand moved towards the left or towards the right, which might not be required in case of identifying smoking gestures.

Finally, for each of the feature vectors derived, we labeled it based on the position on the shelf from where the shopper picked the item (left or right) based on the ground truth information and we used a Random Forest classifier to identify the position of the hand inside the shelf.

5.5 Results

In this section, we present the detailed performance evaluation of I^4S. Our evaluation focuses on three distinct components of I^4S: (a) We evaluate how accurately I^4S can detect picking gestures, (b) We evaluate the performance of the coarse-grained (rack-level) localization process, and (c) We study the fine-grained localization of the pick (shelf-level and within the shelf).

5.5.1 Pick Gesture Identification

As an initial step to identifying pick gestures, we identify the locomotion step of a shopper. To identify the shopper's locomotive state, we extracted accelerometer data from the shopper's smartphone and applied techniques mentioned in [159] and using a 10 fold cross validation, we achieved a precision of 0.963 and a recall of 0.987 in identifying the locomotive state ("stationary" vs. "moving") using a binary classifier. We use this classification model to filter out hand gestures that occur while the user is moving.

Identifying picking gestures: To classify the "pick" gesture, we utilize 2-second

	Accuracy	Precision	Recall
Stationary Store (10 fold cross validation)	92.85%	0.92	0.815

Table 5.5: Accuracy (Precision/Recall) in Identifying Picking Gesture

frames ($w = 2$), with a 50% overlap between consecutive frames. A frame was marked as a true "pick", if it was within ± 2 seconds of the ground truth pick time (marked by the person shadowing the shopper). A random forest-based classifier is trained on the training data (using the smartwatch's accelerometer-based features described in Table 5.3).

Table 5.5 summarises the accuracy, along with the precision and recall of the classifier, based on a 10-fold cross validation strategy. We can achieve an overall accuracy of 92.85%. The precision value is 0.92, indicating that our classifier generates approx. 8% false positives, inferring pick gestures when none was performed. In contrast, the recall is 0.815, implying that we are unable to correctly infer approx. 18% of the actual picks. Note that this performance is based on a person-independent classification model, which does not account for the cross-individual variation in picking activity and other parameters (such as the way the person wears the watch or the hand on which the watch is worn).

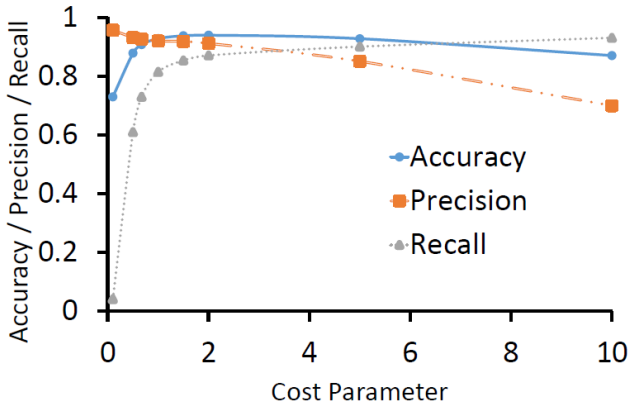


Figure 5.18: Variation of Accuracy/Precision/Recall of 10 Fold Cross Validation for Different Cost Parameter Settings for Picking Being Misclassified

Cost-based classification: Moreover, the system can be tuned to achieve different precision/recall trade-offs –e.g., an application which tries to deliver product-

specific promotions based on the shopper's item-level interactions would desire higher precision (to avoid spamming), while a system looking to identify the general interest level of the shopper (browser vs. interested shopper) may desire a higher recall. Keeping this in mind, we performed a cost-sensitivity analysis of our system. Figure 5.18 shows the system's performance for different cost settings. From the figure we can see that the recall for the system is low for cost less than 1 and it saturates for a cost of 1, while the precision of the system keeps on dropping as the cost is increased. Based on application needs, the appropriate cost setting can be used for the system.

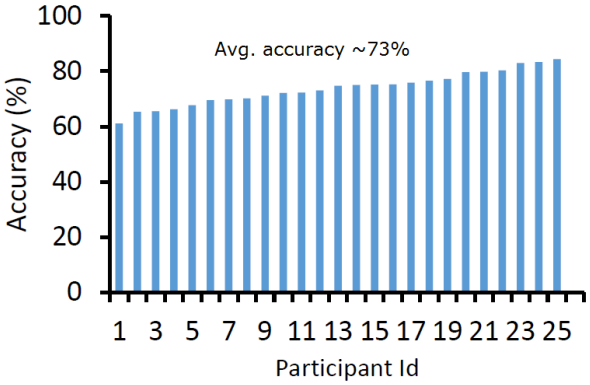


Figure 5.19: Variation of Accuracy Across Users for Leave-One-User-Out Cross Validation

Person independent classification: Finally, we analyse the per-user classification accuracy. Figure 5.19 plots the per user accuracy where each frame is labeled as picking or not-picking. The average accuracy obtained was $\approx 73\%$. However, since picking gesture usually lasts for at least 3 frames (4 seconds), we performed a smoothing of data across 5 frames (extra frames as buffers) and computed the precision and recall for 2 settings - (1) if 2 out of 5 frames were predicted as pick, we considered this to be a pick gesture and (2) if 3 out of the 5 frames were predicted as picks, we identified the gesture as a pick gesture. Table 5.6 shows the performance of detecting a person-independent picking gesture. From the results we can see that tightening the criteria for determining picking (3 out of 5) provides a high precision - i.e. almost 9 out of 10 picks identified for a person are actually picks. However

Smoothing window size	Precision	Recall
2	0.789	0.875
3	0.888	0.674

Table 5.6: Precision and Recall in Identifying Picking Gesture
in a Person Independent Setting with Varying Smoothing Window Length

the recall of the system is low and thus many actual picks are missed.

Application based requirement: Every application can have its specific requirements – e.g. for an application which recommends items to shoppers based on the item that the shopper is picking, it might be acceptable to miss certain picks, but for an application which tries to identify *all* items that are picked when item x is picked by a shopper, it might be okay to have some false positives. Based on the application of the pick-gesture identification, it might be necessary to tune the cost parameters.

5.5.2 Rack Level Location Identification

We next evaluate how accurately we could track the shopper's location in the shop using localization techniques based on the signal (RSSI) heard from the bluetooth beacons heard by the phone. The distance of each of the 778 picks was compared against the 777 other picks. The label of the pick with the smallest distance from the test pick was assigned to the test pick. With this approach, we obtained an accuracy of 58.61%.

Since the number was low, we investigated methods for improvement. The first approach we investigated was - *Count of Beacon Advertisements*. Since there were various losses in the RF signal, we investigated if there is a minimum number of times a beacon should be heard for us to add it to the fingerprint map. We varied the count of number of times a beacon should be heard for it to become a candidate of the fingerprint map. Table 5.7 summarizes the variation in accuracy observed for a basic fingerprint matching technique. From the results we see that even if a beacon

n	1	2	3	5	10	20
Accuracy	58.61%	57.58%	55.78%	57.71%	51.79%	40.74%

Table 5.7: Variation of Accuracy when Fingerprint is Generated Based on Number of Times Beacon is Heard

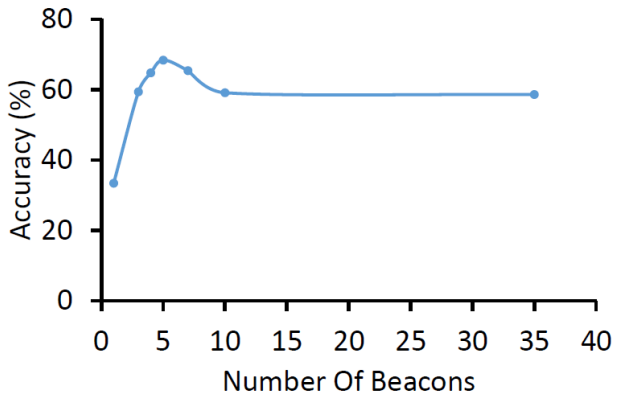


Figure 5.20: Variation of Leave-One-Pick-Out Accuracy for Varied Beacon Count

is heard once, it is useful in localization. Or in other words, since the hearing of beacons might not be reliable, it is good to use any beacon that is heard. From the data evaluation we also observed that there were certain picks which did not hear even one beacon more than 10 times. This prompted us to use any beacon heard in our fingerprint.

We next wanted to understand whether the localization strategy should use the RSSI readings from just a smaller subset of 'stronger-signal' beacons or a wider set of beacons (including ones with weaker signals). For this evaluation, we used the similar fingerprint map as before, but restricted the test beacon vector to the top t beacons, based on the average RSSI heard by the smartphone from those beacons during the pick gesture. Figure 5.20 shows the variation of accuracy when the value of t is changed. From the figure we see that we could achieve the best accuracy (68.63%) if we chose $t = 5$. Surprisingly, if we use $t = 1$ choosing only the beacon with strongest signal strength, the accuracy is quite low. This indicates that a using a technique where location is determined based on the best RSSI heard will fare poorly in such a scenario. For the above setting ($t = 5$), for the location accuracy, we determined the rack level location based on the closest distance ob-

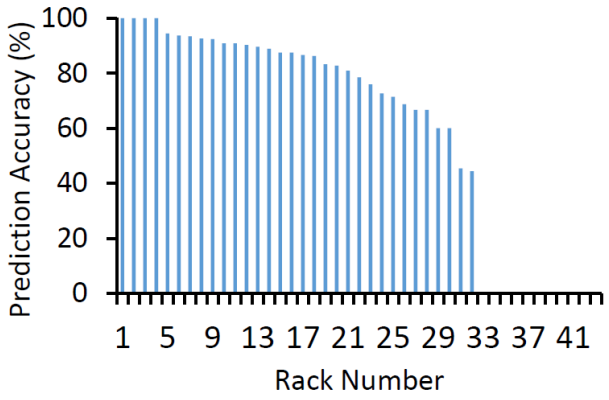


Figure 5.21: Variation of Prediction Accuracy Across Different Racks

served in the fingerprint map. Alternately we also noted the top 3 closest distances predicted and observed that in 80.84% cases, the correct rack was one of the top 3 chosen racks. However using a simple majority voting based technique lowered the prediction accuracy from 68.38% to 61.4%, while a weighted majority voting, where weights were determined based on the distance, the prediction accuracy was 71.2%. Even though the location prediction accuracy was low, having the correct location in the top-k location set indicated that we could use historical knowledge for the localization.

Finally, for the above setting ($n = 1$ and $t = 5$), we performed a Viterbi smoothing. To identify the rack where the shopper is located while picking an item, we used data from the window ($p_0 \rightarrow \pm 2 seconds$ around the pick) where the pick occurred as well as data from three other 4 second windows ($p_{-1} = \{-2, -6\}$ sec, $p_{-2} = \{-6, -10\}$ sec,$p_{-3} = \{-10, -14\}$ sec) immediately preceding the pick instant. We then compute the location probabilities for each window independently and use a depth=4 Viterbi decoding to estimate the pick location. Based on this path-smoothing approach, we improved the rack-level location prediction accuracy to 85.47%.

The location accuracy reported above is skewed by popularity, as the pick dataset will naturally contain a higher number of pick instances for a more popular rack. To understand the un-weighted location accuracy, we computed the pick accuracy

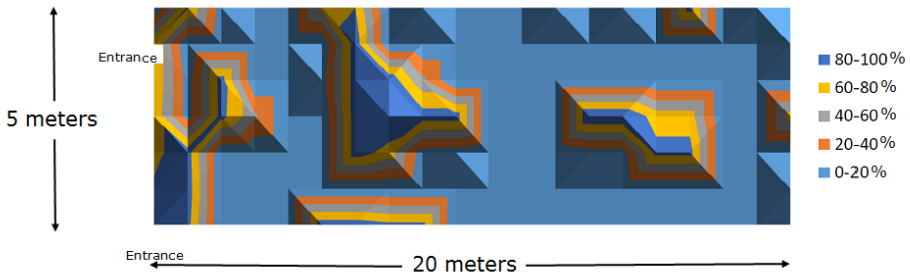


Figure 5.22: Surface Chart Showing Zone Wise Location Prediction Accuracy

for each rack individually and computed the average of these values. The resulting average accuracy was 61%. Figure 5.21 shows the accuracy distribution across the 43 distinct racks. From the figure we can see that 11 racks have an accuracy of 0. On closer inspection, we found that these racks had less than 5 picks in the dataset, indicating that the loss of accuracy was due to lack of sufficient training data. To understand the rack locations with high/low accuracy, we plotted the contour plot of the accuracies. Figure 5.22 shows the contour of the distribution of accuracy. In the figure, zones with no gradient indicates a pathway (some pathways are too narrow to be seen), while the racks with 0 accuracy are indicated by tiny hills with lighter shade of blue. From the figure we can see that the racks in the center had higher accuracy as compared to the ones on the sides, with the exception of the racks near the entrance. The racks at the entrance are the pen stands and was one of the most popular racks in the study. The takeaway here is that the overall location accuracy will be improved via a more carefully-directed data collection phase, where participants are explicitly instructed to make picks uniformly across all racks.

5.5.3 Shelf Level Location Detection

Until now, we have evaluated and identified picking gestures with ≈92% accuracy and the rack level location of pick with ≈85% accuracy. We finally evaluate the performance of our approach in detecting shelves and zones within shelves from where item was picked.

To identify the shelf from where the item was picked, for all the picks in our

Right	Left	← Predicted
111	5	Right
9	60	Left

Table 5.8: Confusion Matrix for Zone in Shelf Identification

dataset, we changed the class label of the picking gestures from picking to the shelf number from where the item was picked. In all we had 6 shelves marked. We performed a 10 fold cross validation on the picks that were identified correctly using the same set of features as before and using a random forest classifier. We found that the accuracy of the classifier in identifying the shelf was 77.12%. On closer analysis we observed that there were a lot of picks which were actually occurring from shelves 5 and 6, but were being labeled as either shelf 1 or 2. A reason for this could be because sitting and picking from a lower shelf might have similar hand trajectory/orientation as standing and picking from an upper shelf. On adding the locomotion state (i.e., discriminating between 'sitting' and 'standing' states) of the user to the feature vector and re-classifying, we found that the classification accuracy increased to 89.07%.

Finally, we wanted to understand whether the item that was picked in a shelf was placed towards the left of the shelf or towards the right. From the data we had observed that picks usually occurred when the person is directly in front of the item of interest. However, in certain cases the person stretches her hand towards left or right to pick the item. From the data we extracted 185 picking gestures (from 22 shoppers' data), where the shopper picked an item that was not directly in front of her, but either to the left or right. Items were picked from multiple shelves and in 116 of these gestures, the shopper picked an item that was towards her right.

We extracted features over a window size of 4 seconds from the game rotation vector sensor data and computed the feature vectors. After labeling the pick as "pick from left" or "pick from right", we performed a 10 fold cross validation using a random forest classifier. Table 5.8 shows the confusion matrix in determining the location of pick within a shelf. From the table we can see that we can achieve an

	Accuracy	Precision	Recall
Person Dependant Picking Gestures Identification	92.85%	0.92	0.815
Person Independent Picking Gesture Identification	NA	0.789	0.875
Rack Level Location Identification (1 meter)	85.47%	NA	NA
Shelf Level Location Identification	89.07%	NA	NA
In Shelf location Identification (0.5 meter)	92.43%	NA	NA

Table 5.9: Summary of the Performance of Various Components of I^4S

accuracy of 92.43% in determining whether the pick occurred from the left half of the shelf or the right.

This shows that our approach could identify the shelf from where item was picked in 89% cases when we used the sensor data from the smartwatch and the smartphone. Within a shelf, if we used the hand trajectory data, we could identify if the item was picked from the left side of the shelf or the right in 92% cases.

5.5.4 Summary of I^4S Approach

Table 5.9 summarises the performance of various components of I^4S. From the table we can see that for every sub-component of the system, the overall performance accuracy is above 85% indicating that it is possible to realise an accurate system which can help in identifying the location from where the shopper picks items. If there exists a separate backend repository that matches individual products with their on-shelf location, a location-based lookup of this repository will directly reveal the specific items (or possible group of items) that the shopper picked up.

5.6 Discussion

Real-world studies show that I^4S is very promising: it can help detect pick events and localize them (to an approximate 3-D location accuracy of 0.5 meters) using the smartwatch and BLE beacons, even in a medium-sized store with narrow aisles and non-regular rack layouts. There are, however, many additional aspects to consider.

Inability to Track Misplaced Items: I^4S's operation is based on the premise that identifying, at shelf-level granularity, the location of a user's pick gesture impli-

citly identifies the product (or product category) selected. While this is likely to be broadly true, store operators know only too well that products are continually being misplaced by shoppers. Hence, if a shopper picks up an item from a shelf where it has been dumped by a previous shopper, the I^4S approach will result in a mis-identification of a shopper's true interest.

Generalisation : The system has been validated in a mid-sized stationary store with students from the university. The user-studies were performed using a LG smartwatch, Estimote beacons and Samsung smartphone. Some of the findings in this study might be environment or device specific. For example, in our study, we found that the magnetic sensor produced erroneous results. However, techniques such as using the magnetometer might be help in improving the accuracy in other settings where the ferro magnetic interference is not high. Similarly, the picks identified in the study are specific to the stationary store. Picking style in other stores (e.g. a clothes store) might be different. This can be validated with additional studies. We are currently expanding to other stores (Details in Chapter 7) to identify in-store differences. Other than the store specific techniques, the devices used might affect the performance. In the feasibility studies we identified that the number of beacons heard by the smartwatch is less than the smartphone. However, we have to perform studies with other devices to determine if the results hold true for various smartwatch brands or if it is specific to the smartwatch used in the study.

The current system has been validated with a specific device type and on students who have similar lifestyle. To generalise the system, additional studies with shopper and device diversity is needed.

Energy Overheads & System Optimization: Currently, I^4S activates continuous sensing of the inertial sensors (accelerometer and gyroscope) and BLE scanning on the smartwatch, primarily because we have no capability to predict the time instants when a consumer may actually pick an item from a shelf. Such sensing obviously drains energy: gyroscope sensing consumes more power than accelerometer sensing, and it is well-known that continuous BLE scanning on smartphones

has less-than-ideal power efficiency [119]. This overhead may not be a serious drawback because of the limited duration of a shopping episode–in our studies, the average shopping episode lasted 6 minutes 52 seconds. However, additional forms of context-driven optimization of such sensing are certainly possible. For example, the user's smartphone sensors may be used to detect when a user is stationary, and turn off the BLE scanning by the smartwatch once the user's location has been established. Likewise, the inertial sensing may also be paused when the shopper is detected at locations that are far away from product shelves (e.g., in non-product areas in large department stores).

Integration of Additional Sensor Devices & Sensors: While I^4S currently uses only the inertial sensors on the smartwatch, a variety of alternative sensing modes may help increase the fidelity level of shopper interaction monitoring. For example, in Chapter 7 we show details of an initial study which utilises a smartwatch-mounted camera to take opportunistic pictures of items being picked, for subsequent image-based product identification. There also exist possibilities of combining infrastructure-based video sensing with I^4S, to improve the accuracy of pick identification and localization. For example, video cameras mounted on either walls or on the top of individual racks may be used to identify the time instants when *a* shopper's hand picks up an item from a shelf, and this time may be correlated with the inertial sensing-based pick time detected by the smartwatch to unambiguously identify *which* shopper picked up the specific *product*. Note that I^4S currently does not directly aim for such product-level resolution, and instead offers shelf-level tracking of shopper interactions.

Device Position: In the current studies, devices had a fixed positions – the smartwatch was worn on the dominant hand, the smartphone was carried in the front pant pocket, while the beacons were placed at the foot of the racks. Experiments were performed to identify the ideal position of the beacons. However, for the smartwatch and smartphone, it was assumed that the smartwatch was worn on the wrist and the smartphone was carried in the pocket. For the smartwatch, every participant

wore the watch on their dominant hand. In future, studies can be conducted to observe the performance of the system, when the watch is worn on the non-dominant hand. In case of the smartphone, the pocket is one of the many positions where shoppers keep the phone while shopping. Since we had provided the smartphone to the participants for the study, they did not perform natural gestures involving the smartphone – e.g. talking on the phone, sending a text message or even browsing. This ensured that the position of the phone was the same throughout the study. However, this is not a natural behavior. In future, studies have to be conducted, where the application is installed on the participant's smartphone. Using the participant's smartphone will help in simulating behavior that is expected in any real world study.

5.7 Summary

Understanding items that a shopper interacted with during a store visit can reveal various interesting insights not only to the store owner or the shopper, but also to social scientists in understanding how shoppers make their shopping choices. In this chapter, I described I^4S– an approach that we have developed to identify shopper's in-store item interaction using multiple sensor data from the shopper's smartwatch and a smartphone. I^4S uses data from BLE scans to determine a shopper's rack level location. Once the scan has determined the person's rack level location, we use the inertial sensor data to determine the shelf level information from where an item is being picked. We can also identify the half from where the item was picked in the shelf. I^4S shows that it is possible to identify '*fine-grained*' '*high-level*' '*ADL specific*' activities using sensor data from multiple devices.

Chapter 6

Understanding Individual's Behaviour

Now that we have established that it is indeed possible to identify daily life activities using multi-modal sensing approaches, we next explore the possibility of going beyond capturing just physical daily lifestyle activities, to potentially understanding higher-level motivations and intentions of individuals during such activities. Unlike *Annapurna* and I^4S, the challenge for this objective is not in inferring the physical activities, such as a shopper's indoor location or their specific gesture. Rather, it is to discover the distinct number of ways in which the same behavioral intent is manifested by properties of the collection of locomotive/gestural states that occur over an entire shopping episode. More specifically, the goal is to extract the meaningful features (over this underlying sequence of locomotive states) that can be used to infer shopper intent. By analysing the sensor traces from multiple personal devices, in this chapter, I show the possibility of determining the behaviour exhibited by an individual during the shopping ADL, even if the shopper has never visited the store previously.

The anecdotal observation that – *an individual who is in a hurry and needs to buy an item, will move quickly through a store, pick the item and check out. Comparatively, an individual who is not in a hurry, has the liberty of spending some time brow-*

sing through items not in her shopping list – serves to illustrate the point that some aspects of a shopper's psychographic profile may be revealed from his or her physical actions. Through *CROSDAC*, a novel non-person specific approach, we show the possibility of identifying an individual's shopping intent by analysing these physical actions. *CROSDAC* utilises sensor data from the smartphone's inertial sensor as well as Wi-Fi scans to determine these physical actions. Inertial sensors provide information about an individual's locomotion state (sit,stand,walk,turn), while the Wi-Fi scans provide location details of the individual.

There are two major challenges that *CROSDAC* has to address: (a) the same intent can be manifested in diverse ways – e.g. a hurrying 15 year old might have different physical signature from a hurrying 70 year old, and (b) the number of such manifestations is not known apriori. For the first problem, prior research suggests that the diversity is directly influenced by demographic attributes, such as height, weight and gender (e.g. [63]). However, since its not know which all factors influence shopping behavior, we propose *CROSDAC*, an unsupervised technique to determine the number of diverse manifestations – we call this the "shopping style". Once *CROSDAC* determines the shopping style, it identifies the intent based on similarity of the shopper with others exhibiting similar shopping style. We have tested other variations of *CROSDAC* and found that the aforementioned approach could best identify a shopper's intent.

Even though identifying behavior using sensor data of a smartphone has been studied by various researchers(e.g. – [103]), we believe that the *CROSDAC* approach is the first one targeting behavioral identification through non-invasive mobile sensing in the retail domain.

6.1 Necessity of Identifying Shopping Behaviour

Before discussing the design of such systems, let us understand the impact of understanding a shopper's behavior. Consider the following scenarios:

Scenario 1: *Alice walks into a store looking for a pair of jeans. She spots the appropriate section, selects her favorite pair and heads towards the check out counter. However, before reaching the checkout counter, she spots a beautiful top that would go well with the jeans and stops to look at it. For a while she considers buying the top too, but when she sees the price tag, thinks for a while and decides that it was above her stipulated budget and goes ahead with just the jeans purchase.* This is a scenario that many of us might have experienced. Now reconsider the scenario - *While Alice was admiring the top, a backend system determined that Alice is interested in purchasing the item. However, as soon as she flips the price tag and starts reconsidering whether to make the purchase, the system determines that Alice is confused, but with desire to purchase the item. The system takes the active decision to send a promotion for the top to Alice to tempt her in making the purchase.*

Scenario 2: *Bob, the store assistant observes the two customers shopping in his retail store. While customer 1 is wandering around the store, customer 2 stands in front of a rack for a while and then moves to another rack and then the next. Assuming that customer 2 needs assistance, Bob goes ahead and starts talking to the customer. It turns out that the customer 2 is frittering away his time and has no buying intention. It also turns out that the customer is extremely chatty and for courtesy's sake, Bob is not able to cut short the small talk. At some point Bob notices customer 1 leaving the shop and wonders if she actually needed assistance.* In this scenario, it would have worked wonders for Bob if he had knowledge of each customer's intentions, even before he approached any of them. In case he found that customer 1 was confused and customer 2 had no buying intention, he would have assisted customer 1 and the assistance might have resulted in a sale.

Even though the above scenarios might sound like a far fetched idea, but with the availability of multiple sensing devices and information from similar shopper's exhibiting similar behavior, I believe that the above-described scenarios will become a reality in the near future. In this chapter, we design and evaluate one possible

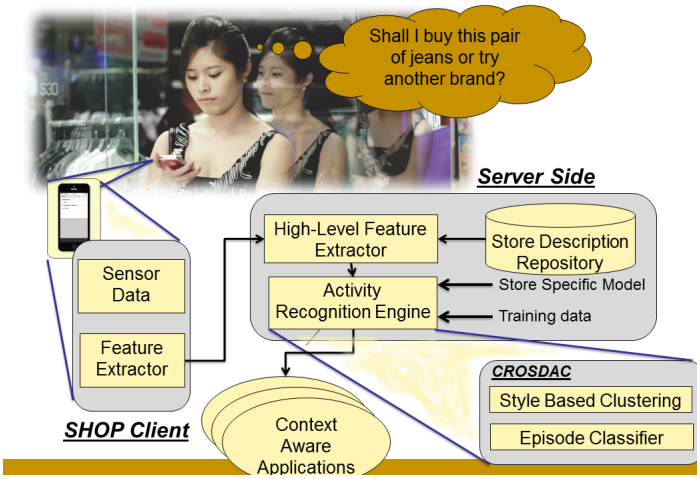


Figure 6.1: *SHOP* Overall Architecture

approach towards determining a shopper's intent.

6.2 System Overview

Keeping the above scenario and all associated challenges in mind, we explored the possibility of identifying the behavior of the shopper using sensor data from her personal devices. Assuming that the contextual knowledge of the shopper – her trajectory using Wi-Fi / BLE localization and sequence of performed activities using phone's inertial sensors – could be extracted from her smartphone through "*SHOP*" a store specific application, we explore techniques to determine the shopper's exhibited behavior. We call this behavior determination component *CROSDAC*. In this section, we describe the overview of the store specific application - *SHOP*, that could run *CROSDAC*. Figure 6.1 illustrates the client-server architecture of *SHOP*.

The shopper's smartphone (and her wearable devices) runs the *SHOP* client, which collects the raw sensor data from the phone and extracts relevant sensor features from such data. In our present smartphone-based implementation, the sensors used include the accelerometer (to perform micro-activity recognition) and the Wi-Fi sensor data (i.e., the RSSI readings from different APs heard by the smartphone). This data is used by the *CROSDAC* approach.

At a high-level, the *CROSDAC* approach focuses on organically separating out the training data into a relatively small set of distinct shopping styles, and building separate classifiers for each style. The rationale behind such separation is our belief that different segments of a crowd-scale population do manifest the same intent in fundamentally distinct ways. The central principle of our *CROSDAC* approach is borrowed from the speech recognition domain, where it is well known that words are better classified once they are grouped by speaker accents–i.e., if separate recognition systems are built for each distinct accent [15]. *CROSDAC's* design rests on our belief that shopping too has such hidden *accents*, which if captured, can help us better classify individual shopping behavior. We refer to this analogous concept as the *Shopping Style* of the shopper. Examples of such styles might be a tendency to look through various items on display in a deliberate fashion first before narrowing down focus on a specific brand, or an inclination to first do an overall reconnaissance of a store's entire floor area before then focusing on the sections of primary interest.

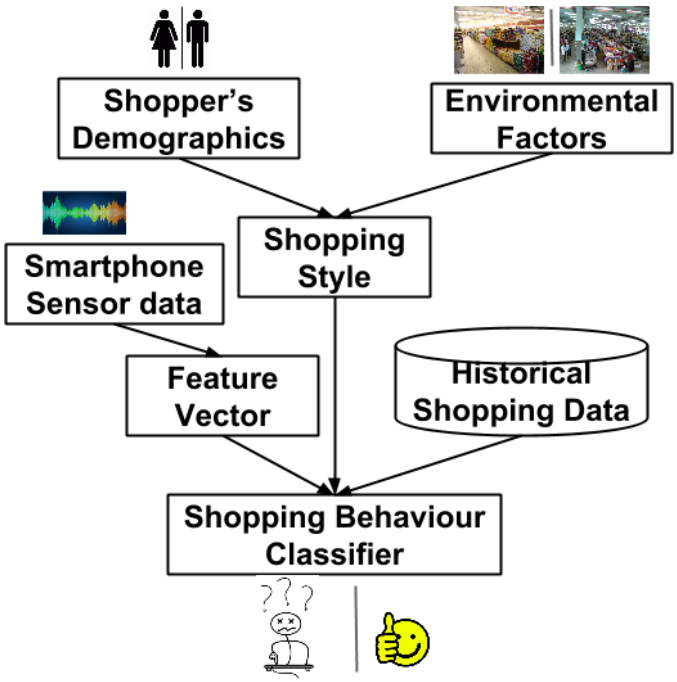


Figure 6.2: Steps in *CROSDAC* Classification

We believe that such *styles*, if they exist, are a hidden or *latent* property of a

shopper, which is dependent on a combination of factors such as *demographic attributes* (e.g., age, gender, ethnicity), *lifestyle attributes* (e.g., food preferences, level of disposable income) and exogenous *environmental attributes* (e.g., the crowdedness of a store or narrowness of the shopping aisles). Figure 6.2 illustrates our representation of this cause, effect and observed behavior. The *CROSDAC* approach is to *somehow* capture the "distinct shopping styles" that are present in the crowd-scale shopping population, and then have the actual classification process be *moderated* by these distinct styles. As shown in Figure 6.2, the approach is to (a) discover shopping styles without explicit apriori enumeration of such styles, and then (b) have the shopping styles influence the classification logic (which takes as inputs the smartphone-generated sensor features, potentially historical shopping episodes and these uncovered shopping styles). Intuitively, this is achieved through some form of *clustering*, prior to the step of classification, with each cluster representing episodes with the same *shopping style*. The question then is: can we, as in [63], form such clusters simply based on demographic/environmental attributes, or is it better to discover such clusters through other *unsupervised* means? To investigate this question, we shall outline some of the possible alternative techniques (involving classification and clustering) that *CROSDAC* may utilize.

In order to describe the alternative techniques precisely, let us first define some mathematical notation. Assume we have sensor traces from m users $M = \{1, 2, \cdots, m\}$, each performing n shopping episodes (in our scenarios, n may even be 0, corresponding to cases where the user visits the store for the first time. Let p of these $m \times n$ episodes have a shopping behavior label, i.e., they constitute the *training data*. Let (D) be the demographic and (E) be the environmental factors associated with each of them. Our goal is to predict the shopping behaviors of the remaining $m \times n - p$ using the p labeled episodes as the training set.

We map the sensor traces into a k-dimensional feature vector F, where $F_{ij} = \{f_1, f_2, \cdots, f_k\}$ denotes the feature vector corresponding to the j^{th} shopping episode of the i^{th} user. The choice of these features can, as mentioned, vary by location.

If we denote the set of shopping behavior labels as $B = \{b_1, b_2, \cdots, b_l\}$, predicting the shopping behaviors of the remaining $m \times n - p$ using the p labeled episodes as the training set can be formulated as learning a concept function c_i for every individual i such that:

$$\forall j, c_i(F_{ij}) = b_x \text{ where } b_x \in B.$$

Now that we have introduced the mathematical notations, let us consider some of the possible approaches for *style-aware* non-individualized classification are:

- *Single-Level Supervised Classifier (U1)*: This is a supervised classifier (e.g., decision tree) that uses all the p labeled episodes as training data and learns a single classification model c for all users. As and when a new episode F_{ij} comes in, the episode is passed through the supervised classifier, which classifies this episode into one of the shopping behaviors in B, i.e., $\forall i, j, c(F_{ij}) = b_x$.
- *Unsupervised Clustering and Supervised Classifier (U2)*:In this approach, the feature vectors of all the p episodes are clustered using a distance function d such that, any two feature vectors F, F' are put into the same cluster only if $d(F, F') < s$, where s is a threshold. All episodes inside a cluster are taken along with the episode label and a supervised classifier is created inside each cluster. To classify an unlabeled F_{ij}, it is first put into a cluster whose center is at a minimum distance from F_{ij}, and subsequently classified using the cluster-specific classifier.
- *CSN like approach (C1)* [63]: Using the concept of homophily, a similarity-based supervised classification technique is built. This model assumes that people who are similar physically (e.g., gender) or in lifestyle (food preference or frequent visitor of the store) will behave in similar ways. Multiple supervised classifiers, one for each kind of similarity (male-classifier, female classifier, etc) are built from the p labeled episodes ($Classifier_{male} =$

	U1	U2	C1	C2
Applies clustering prior to classification	N	Y	N	Y
Builds a separate classification model for each demographic attribute	N	N	Y	Y
Requires knowledge of demographic labels for each episode	N	N	Y	Y
Fuses predictions from multiple *demographically-filtered* classifiers	N	N	Y	Y

Table 6.1: Various Approaches for Crowd-Scale Shopping Behavior Prediction

$\{F_{ij}|D_j \in \{male\}\}$, $Classifier_{female} = \{F_{ij}|D_j \in \{female\}\}$, etc). When a test episode comes in, it is assumed that the demographic and other personal details of the user is known and the classifier's corresponding to the corresponding values for the demographic/lifestyle attributes are chosen–e.g., a male, vegetarian shopper will be classified by both the $Classifier_{male}$ and $Classifier_{veg}$ classifiers. Each classifier predicts the class label with a certain confidence. If $\{t_1, t_2, \cdots, t_T\}$ are all the classifiers built and $conf_{it}$ is the prediction confidence of classifier t for the behavior b_i, then the overall prediction for the episode is determined by $\max_{1 \leq b_i \leq B_i} (\sum_{t=1}^{T} conf_{b_i}) where t \in D_i$.

- *Unsupervised Clustering and applying CSN like approach(C2)*: This approach is effectively a combination of *U2* and *C1*. Here, all the episodes are clustered into multiple smaller clusters;subsequently, for each cluster, T separate classifiers (corresponding to each attribute value) are built, as in *C1*. When an unknown (test) shopping episode comes in, it is first assigned to one cluster (applying the clustering logic of *U2*), followed by an overall prediction computed by combining the confidence of the multiple cluster-specific classifiers.

Table 6.1 summarizes these approaches, listing the important ways in which they differ, both in terms of their processing steps and the assumptions that they make about the incoming shopping episodes.

6.3 Design Choices

To gain insights into the real-world feasibility of using *CROSDAC* approach to analyse shopping data to infer shopper's intent, we had to determine the number of shopping styles that existed in the data. We took an empirical approach, where we used the data to determine the number of shopping styles that existed in the data. To understand the generalisability of *CROSDAC*, we performed two user studies. In this section, I first describe the two distinct user studies which we conducted. While the first study was conducted in a food court located in a large shopping mall in New Delhi, the second study was conducted in the University's gift shop in Singapore. The description of the user study is followed by the details of the approach taken by *CROSDAC* to determine the number of "shopping styles". Based on the "shopping styles", *CROSDAC* determines the overall shopping behavior/intent. Even though the overall objective of both the studies was the same, the setting in the two studies were quite different in terms of location, type of store, size etc. Table 6.2 provides a summary of the two studies, highlighting some significant differences between the two studies.

Cognitive state/ behavior identification: To derive insights into the different behaviors exhibited by shoppers during a store visit, we surveyed relevant literature on consumer behavior and marketing. The shopping behavior identification literature indicates that shoppers can exhibit several behaviors during a store visit. At a high level, the shopper might be in a store with purchasing intentions or might be browsing [19]. We term this category of shoppers who are browsing as shoppers with *no buying intention* ('NBI'). Amongst the shoppers with purchasing intentions, there can be several sub-category (e.g., shoppers were categories into 8 sub-categories in [155]). One common sub-category which we identified in several articles, including in [155] was *confused by over choice*. This influenced us to explore the possibility of determining if the shopper is confused (we term this category as *has buying intention, but confused shopper* ('BIAC')) using sensor data

	Study 1	Study 2
Location of the Study	Food Court in a shopping mall in New Delhi	University's gift Store in Singapore
Number of participants	30 (15 males, 15 females)	22 (12 males, 10 females)
Size of Location	Large - housed multiple stores	small – $\approx 50m^2$
Devices used	Samsung S II smartphone, Wi-Fi AP	Samsung S IV smartphone Wi-Fi Access Point LG Urbane smartwatch

Table 6.2: Summary of the Studies Conducted to Understand Shopper's Behavior

from the shopper's personal devices. The category which represents the shoppers who are not confused (BIAC) is the focused category. We termed shoppers in this category as *has buying intentions and is focused* ('BIAF') shoppers. In addition to identifying the BIAC category shoppers, we also explored the possibility of identifying NBI and BIAF shoppers through sensor data analysis. This behavior based categorisation of shoppers is similar to the *influence of attitude on purchase* shown in [154]. In addition to the above mentioned behavior based categorisation, literature also indicated that social factors – like shopping in a group affects a person's shopping [139]. This motivated us to explore the possibility of identifying the influence of social factors on shopping. We termed shoppers in this category as *buying intention and in group* shoppers ('BIG').

6.3.1 Datasets

6.3.1.1 Study 1: Food Court in New Delhi

The first study was conducted in the food court of a large shopping mall in New Delhi. For the study, we recruited 30 distinct volunteers (15 males and 15 females) from amongst the shoppers. The participants were made aware that they would be asked to perform certain tasks, where each task was considered as an "episode". Altogether, 86 "episodes" were collected from the 30 shoppers over 14 days in the food court area (next to the movieplex) of the mall. Figure 6.3 provides a schematic of food court's layout, which consisted of 12 F&B stalls and 2 centralized

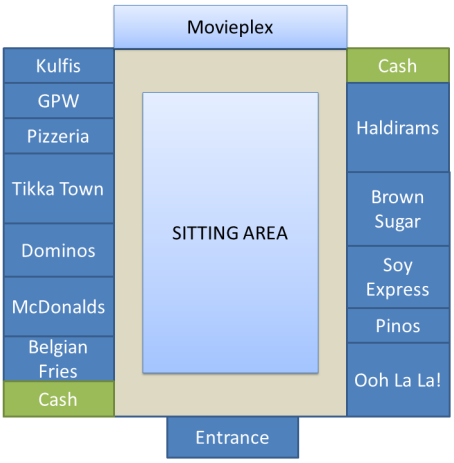


Figure 6.3: Schematic Layout of Mall

cash counters for buying coupons (purchases in each store required redemption of coupons). The central area of the food court had seats, where the person could sit and consume her meal. Even though the mall had retail stores, our justification of conducting the first study in the food court was (i) it is a semi-public area, and thus easier to perform experiments without requiring the consent of individual retailers, and (ii) the adjacency of the food court to the movieplex meant that it could provide us with the necessary diversity of behavioral intent.

To mimic different types of shopping behavior, the participants were asked to perform certain semi-guided tasks to simulate different shopping behaviors, without any prescribed time limit. The tasks belonged to four categories, corresponding to 3 distinct types of shopping intent/cognitive states:

1 *No Buying Intention (NBI)*: This label captures participant behavior when she was unlikely to make a purchase. To generate such behavior, NBI users were instructed to "Please wait here for a friend joining you for a movie".

2 *Buying Intention-Alone-Focused (BIAF)*: This label captures behavior when the participant has a premeditated purchase goal. This behavior was generated through instructions such as "Please buy a cold beverage for you and a friend" (note that beverages are available at multiple stalls, implying that the participant still had to exercise some amount of choice.)

3 *Buying Intention-Alone-Confused (BIAC)*: This label is intended to capture

behavior where the participant has a purchase intention, but the precise nature of the purchase is fairly ambiguous. To generate such behavior, BIAC participants were instructed to "Please buy a small-sized meal for your mother-in-law".

4 *Buying in a group (BIG)*: This label was designed to capture the buying behavior of participants when they shopped in a group; BIG participants were told to "Please help your friend (who was with the shopper) in choosing what to eat".

While our initial goal was to study both individual and group-level behavior, we realized that individual behavior itself is challenging to analyze. Hence, we do not consider the group-interaction based episodes further in this dissertation and confine our studies to NBI, BIAF and BIAC participants. The resulting dataset had 67 episodes (20 NBI; 19 BIAF and 28 BIAC). The average episode time for the 67 episodes was 520.59 seconds with a standard deviation of 242.49 seconds. For the 67 episodes which we considered in our study, 7 participants performed tasks corresponding to all three categories, whereas 23 participants performed tasks corresponding to any two of the three categories.

It must be noted that the participants were given high-level behavioral tasks which they enacted. There was no physical restriction imposed on the participants while they completed the study. The participants were neither asked to follow any specific physical activity (e.g. walk in a certain route or sit down after every one minute), nor was any time factor limitation imposed in the study (participants were free to take as much time as they desired).

Sensor Data Collection– To collect the mobile sensing data, each participant was provided with a Samsung SII phone, which had a pre-installed application running and continuously collected sensor data. The participants were asked to carry this phone in any pocket of their clothing. The application captured (i) accelerometer and compass data, which was later used to identify four locomotion states (sit, stand,walk and turn) exhibited by the participant, and (ii) Wi-Fi scan data, which

was later used to compute of location of the participant in an indoor environment using the Horus algorithm [163]. The locomotion state was calculated every 5 seconds and the trajectory state was computed every 15 seconds.

Ground Truth Collection– Other than collecting the sensor data from the phone, we also recorded the "ground truth" (including the locomotive actions and the location trajectory) for each of the shopping episodes. This ground truth was collected by an observer, who *shadowed* the participant and noted down their various activities, using a separate custom Application (similar to the shadowing application described in Section 5.3) for recording user behavior.

6.3.1.2 Study 2: University Gift Shop in Singapore

The second user study was conducted in our university gift shop – a small sized souvenir store. Recruitment for the second study was done through word-of-mouth information passing to members of the University. In all, we recruited 22 volunteers, of which 12 were males and 10 were females. 19 of the 22 volunteers were undergraduate students in the university, while 1 was a post graduate student and 2 were staff members of the university. Similar to the previous study, each participant was asked to carry out certain tasks. The tasks belonged to the three categories – confused (BIAC), focused (BIAF) and no buying intention (NBI). Each participant executed 3 tasks (one from each category) and in all we collected 66 "episodes", but due to error in data collection, we had to ignore data from all 3 tasks of one user and 1 task each from two other users. So finally, we used 61 episodes for our data analysis.

The entire study was carried out over 5 days. Figure 6.4 provides a schematic layout of the study venue. Unlike the distinct shops in the Delhi food court, the gift shop did not have any well defined zoning. However, I have marked the zones in the shop based on the majority items that were present in the zone and these zones have been referred to as shops while extracting features. It must be noted that the zoning in the figure did not define a pure zone; some of the clothing zones had gifts

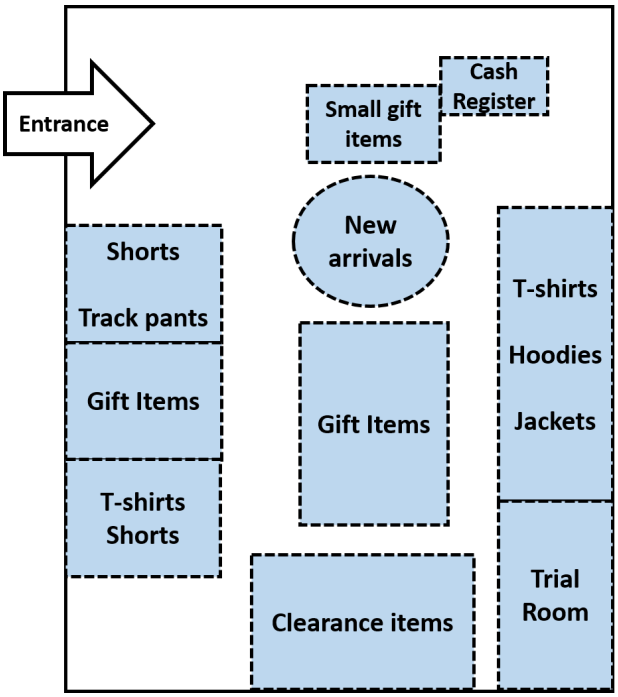


Figure 6.4: Schematic Layout of University Gift Shop

and vice versa.

Similar to Study 1, for this study, participants executed some semi-guided tasks to mimic shopping behaviors and the broader category was similar – the shopper had to execute tasks mimicking focused, confused and no-buying-intention. However, since the location of the study was not a food court, the details of the tasks varied.

1 *No Buying Intention (NBI)*: To generate a no-buying-intention behavior, users were instructed to "Spend some time in the shop window shopping before your friend joins in".

2 *Buying Intention-Alone-Focused (BIAF)*: For the second study, this behavior was generated through the instruction similar to "buy a t-shirt for yourself". Again, to ensure that the buying required some choosing, the item that was suggested in the instruction had multiple options – e.g. multiple t-shirt designs.

3 *Buying Intention-Alone-Confused (BIAC)*: To generate the confused behavior in the gift shop, participants were instructed to "Please buy a gift item for an acquaintance".

Similar to the previous study, the participants in this study were instructed to follow the behavioral tasks. There was no restriction imposed on the physical shopping activity or the time taken to carry out the entire shopping. Shoppers were free to move as they wished and look at/pick items from any rack in the store.

Sensor Data Collection– In Study 1, we collected sensor data using the Samsung S II smartphone. In this round of study, we provided the Samsung S IV smartphone to the participant. The participant also wore a LG Urbane smartwatch (note: smartwatch data has not been used in the behavior analysis). The participants were asked to carry the smartphone in the front pocket of the clothing and wear the smartwatch in the dominant hand. Both the devices were running our custom application, which collected not only the inertial sensor data and Wi-Fi scan data, but also ble scan information. Again, similar to study 1, we used inertial sensor data from the smartphone to determine locomotion state, while the location of the shopper was determined through the Wi-Fi scan information. Even though we collected BLE information, the location derived in the final evaluation was through Wi-Fi based localization using the RADAR technique [9].

Ground Truth Collection– The ground truth data collection in this study was done by shadowing the shopper, in a manner similar to Study 1.

6.3.2 Determining Number of Shopping Styles

Since we believed that the data from shoppers could be clustered based on shopping styles, we had to divide the data into clusters. However, we did not have before hand knowledge of the number of existing shopping styles; thus, we first estimated the number of clusters (or shopping styles) in the shopping episodes. For this determination, we used an empirical approach. Since the number of clusters could vary across studies, we used the data from each study to determine the optimum cluster size for the particular study.

To understand the likely shopping styles embedded within our observational

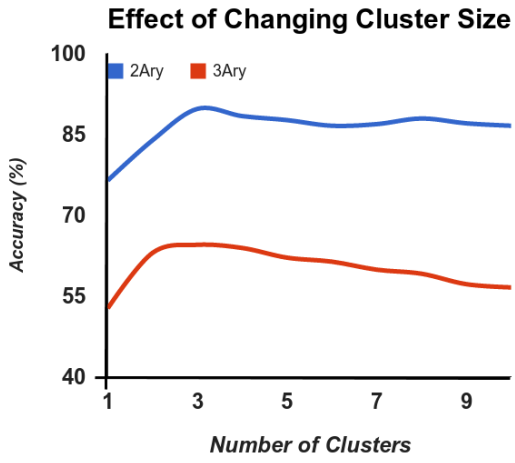


Figure 6.5: Study 1: Effect of Cluster Size on Prediction Accuracy

data, we studied the accuracy of the best-performing *U2* approach (details in Section 6.5) by varying the number of clusters K specified in a simple K-Means clustering algorithm [42]. Accuracy for a value of k was defined as the sum of all correctly predicted instances in every cluster divided by the total number of instances in the entire dataset. We next see how the accuracy is affected by the choice of K.

6.3.2.1 Determining K in Study 1

For study 1, we used the ground truth data to determine the variation in accuracy. Figure 6.5 presents the variation in classification accuracy for different values of K. From the figure we can see that $K = 3$ provided the best performance for *2-ary* classification, while the performance of both $K = 2$ and $K = 3$ are similar for *3-ary* classification. To make a conservative estimation, we decided that there were three latent shopping styles in our dataset and used $K = 3$ in all our studies. An observation we made in this analysis was: for higher value of K (greater than 4) certain clusters evolved with less than 3 episodes indicating that the clustering algorithm was looking too narrowly for clusters and we were running into the danger of splitting one shopping style into multiple clusters if we chose larger cluster sizes.

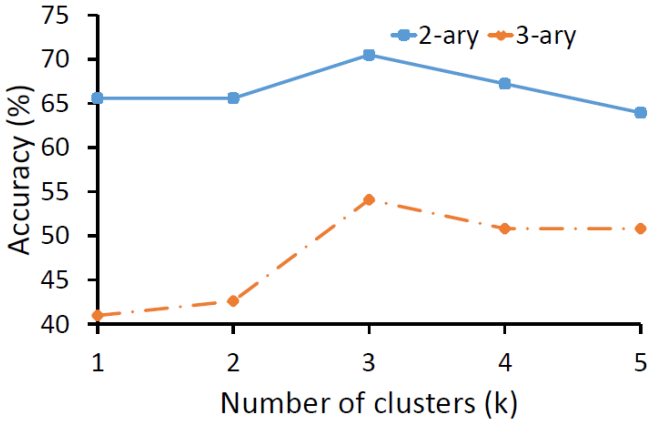


Figure 6.6: Study 2: Effect of Cluster Size on Prediction Accuracy

6.3.2.2 Determining K in Study 2

For the second dataset, similar to study 1, we tested the variation of accuracy for different values of K. Figure 6.6 presents the variation of accuracy for different values of K. From the values we can see that again the best performance is at $k = 3$ for both 2-ary as well as 3-ary classifier. Again, similar to the previous study, in this study also we see that as we increase the number of clusters, there is a dip in accuracy after an initial rise. This is a strong indication that even though clustering might improve in identifying shopping styles, having too many clusters isolates certain episodes.

Based on these explorations, we used $K = 3$ as the number of clusters in both our studies.

6.4 Methodology

We now investigate an approach for recognizing various abstract aspects of in-store shopping behavior from an individual's mobile/wearable sensor traces. The *CROSDAC* approach seeks to (a) first identify these distinct styles in a data-driven fashion (from the underlying multi-user training data), and then (b) have these styles moderate the actual classification process. The identification of styles is performed

via a clustering technique, whereas a separate classification model is then developed from training data for each specific cluster. For all our experiments, we used Weka [41]'s implementation of (i) k-means algorithm for clustering (ii) J48 decision tree for classification and (iii) 10-fold cross validation strategy.

Steps in the data processing pipeline included: (1) extracting sensor data from the smartphone and framing them. The sensor data included accelerometer data as well as the Wi-Fi scan results. (2) extract micro level details - instantaneous micro level activity and location from the frame (3) use the micro level details to extract features for an entire episode (4) Cluster episodes with similar shopping styles together (5) Apply a classifier to determine the shopping intent.

6.4.1 Feature Vectors & Classification

For an entire episode, we had a feature vector of length 25. We next describe the features that were calculated at the third step of the data processing. The features were extracted from the micro level locomotion and location details. The features were divided into two classes:

- *High Level Locomotive Features:* These are obtained from (accelerometer, compass) readings. These include ($f_1 \cdots f_8$) - number of [sit; stand;walk;turn] frames and percentage of time spent in each of these activities, and (f_9) - the number of state transitions.
- *Trajectory Features:* These are obtained from Wi-Fi based location traces. The trajectory features is broken into two levels:
    1 *Grid level features*, which consists of location at grid – level granularity, both at (i) micro-level (shopping area broken down into 10x10 grids) and (ii) macro-level (shopping area broken down into 2x2 grids). Micro-grid level features included (f_{10}) - number of grids visited at least once, (f_{11}) - total number of grids traversed and (f_{12}) - the number of re-visits to an individual grid, where the count is incremented if the user visits a grid

she had visited previously. Macro-grid level features included ($f_{13} - f_{16}$) - the percentage of time spent in every grid, and (f_{17}) - the grid wise entropy calculated from the proportion(P_i) of time spent in each grid using the formula $-\sum_{i=1}^{4} P_i * log(P_i)$. Repeated visits to grids and high number of grids traversed indicated either a BIAC or a NBI. At a macro level, when a user spent most time in one grid, it was indicative of a BIAF user.

2 *Semantic level features*, i.e., computed at shop-level granularity included (f_{18}, f_{19}) - the number of shops visited (both repeating and unique) , (f_{20}) - highest time spent stationary in front of a shop, (f_{21}) - total time spent in shops, (f_{22}) - mean time spent in shops, (f_{23}) - Standard deviation of time spent in shops, (f_{24}) - the total episode time and (f_{25}) - proportion of time spent in top shop i.e (f_{20}/f_{24}). If the person spent a high percentage of her episode time in front of shops, it was often indicative of BI; likewise, if the difference of number of shops visited and the number of unique shops visited was high (indicating multiple visits to the same shop), it indicated BIAF user.

6.5 Results

The above mentioned features were used to determine every shopping behavior. For each of the studies, we determine the accuracy of *CROSDAC* and our insights of the shopping behavior. We investigate how the various methodologies described previously perform in identifying the shopper's behavior. We investigate two cases here: (1) 2-Ary classifier which tries to identify whether the shopper has buying intentions (BI) or no buying intentions (NBI), and (2) 3-Ary classifier which tries to distinguish across all 3 labels: BIAF, BIAC and NBI. For the 2-Ary case, both BIAC and the BIAF labels map to the BI class label. Using a 10-fold cross validation methodology, we investigate the resulting classification accuracy.

		U1	U2	C1	C2
Sensor Data	2-ary	71.6	77.6	64.7	71.47
	3-ary	46.02	52.2	41.3	42.05
Ground Truth	2-ary	76.41	89.7	74.77	83.7
	3-ary	52.68	64.47	45.9	56.57

Table 6.3: Study 1: Classification Accuracy for Sensor Data and Ground Truth

In this section, we also scrutinise the types of episodes that fall in each cluster and finally, we identify the features which have the highest distinguishing ability.

6.5.1 Study 1: Food Court

For our studies, other than the sensor traces, we also had the data from the ground truth. For the data analysis, we computed the performance of both: (1) the sensor trace which is obtained from the participant smartphone, and (2) the ground truth obtained via shadowing. The difference in the performance between the two traces would help us understand the magnitude of inaccuracy that creeps in because of the inaccuracy in determining micro- features, which might either be the locomotive feature or the instantaneous location.

6.5.1.1 Classification Accuracy

Table 6.3 shows the performance of the different techniques for data obtained from the sensor trace as well as the ground truth. From the results we can see that the identification accuracy in case of ground truth based analysis is much higher than the sensor trace information, with accuracy difference being as high as 10% in certain cases, showing the importance of identifying the micro-features more accurately. However, since we have to determine the episode type from the sensor data, we scrutinise the sensor trace to identify some key findings. There are some interesting findings from the data:

- For the sensor trace data (as well as the ground truth data), we find that the clustering-based algorithms(U2,C2) outperform their respective basic coun-

terparts (U1,C1), with the pairwise accuracy gains exceeding 10% (for U2 vs. U1). As clustering is our implicit method for identifying latent shopping styles, the results suggest that identifying and using such styles as a basis for differentiation is central to robust performance.

- Explicit use of demographics as a basis for clustering similarity is not always beneficial. Note that U1 outperforms the corresponding demographics-aware, CSN-equivalent, method (C1).

- Finally, note that the classification accuracies for our style-based approach, U2, is higher than all the other approaches (Quite high if ground truth data is used). This suggests that mobile sensing-based classification of such abstract shopping attributes may indeed be possible-such classification will become more accurate as wearable sensing becomes more commonplace.

6.5.1.2 Error Analysis

We next wanted to understand which classes were being confused for another. When we compared the prediction vs actual class label for each episode, we found that identifying NBI was easier to detect as compared to the other classes. We take an observational approach and identify characteristics which might have caused the error. For the NBI task, we observed that participants commonly used their cellphone (to make a call, use an App or play games) while sitting in the central area or walking around the food court. Some NBI participants ambled around the food court, glancing through the menus of the stores. Comparatively, BIAF participants usually either (i) went directly to a beverage-carrying store right next to the cash counter after purchasing their cash card, or (ii) went to 2-3 stores before making their purchase (from post-task interviews, this was attributed to either longer queues or their preferred drink being unavailable at the initial stores), while BIAC participants also visited multiple stores, but their browsing time in front of each of the stores was much higher. BIAC participants exhibited two types of confusion: (i) inter-store,

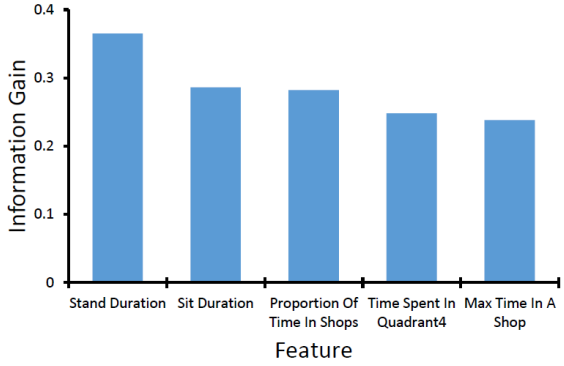


Figure 6.7: Study 1: Features with the Highest Information Gain

where the participant was unsure about the store selection, and (ii) intra-store, where the participant is unable to choose from the menu within a store. When we analysed the confusion matrix for the study, we found that the intra-store confusion behaviour was sometimes perceived as focused behaviour (as both intents exhibited a long period of being stationary at a single store). From the post-survey, we also discovered a cultural aspect: vegetarian participants preferred going to stores selling only veg food items. While our observational dataset is quite small, similar cultural traits are likely to exist in other geographies (e.g. people going to Halal food stores).

6.5.1.3 Information Gain

Finally, we wanted to understand which features had the highest influence in predicting the class labels. For this, we ranked the features based on their information gain. Figure 6.7 shows the 5 features which had the highest information gain. From the list we can see that duration for which a person stands or sits is a key feature in this identification. Since a shopper who made a purchase stood near the food court counter for a while, this feature played an important role. Similarly, a person who had no buying intention would take a seat in the sitting area of the food court. Other features which ranked highly in the information gain list are: proportion of times at shops - which had a strong influence in filtering no buying intention and max time in a shop: again indicating that the person was making a purchase.

	U1	U2	C1	C2
2-ary	65.57	70.49	65.57	65.57
3-ary	40.98	54.09	42.62	47.54

Table 6.4: Study 2: Performance of Different Approaches in the University Gift Shop

6.5.2 Study 2: University Gift Shop

We next study the performance of the various techniques in predicting shopper's behavior in the University gift shop.

6.5.2.1 Classification Accuracy

Table 6.4 shows the performance of the different techniques obtained from the sensor trace. From the table we can see that (i) Similar to Study 1, in Study 2, the U2 performs better than U1 for both 2-ary as well as 3-ary. But for the CSN like approach, we found that in case of 2-ary, the performance of both the approaches C1 and C2 are similar. This indicates that the performance of clustering based identification approach is at least at par with the non-clustering based approach, if not better. (ii) Overall U2 again has the highest accuracy and in this study, for 3-ary classification, the performance of U2 is much higher than its counterpart U1. However, the accuracy of U2 in this study is lower than the accuracy obtained in Study 1. In Section 6.6, we discuss about the effect of the environment on the performance.

6.5.2.2 Error Analysis

Similar to the previous study, we analyse the possible error causes in the 3-ary prediction. For this study, we found that almost all the classes had similar true positives (between 50 to 60%), with BIAC being the highest. In terms of actions observed, in our studies we had some observations which demonstrates behavioral difference in diverse store types. We first describe the behaviors observed in the study. For the focused task (BIAF), many participants went directly to one zone selling t-shirts, chose one t-shirt and returned to the checkout counter. However, there were some

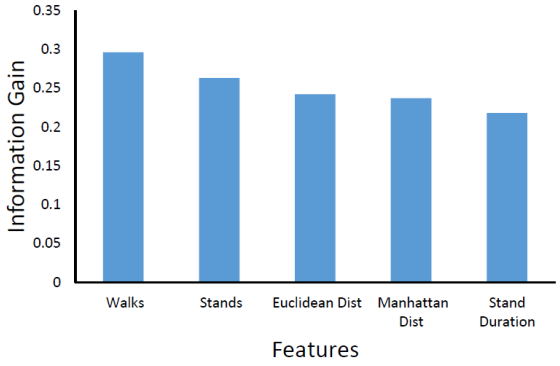


Figure 6.8: Study 2: Features with the Highest Information Gain

participants who moved back and forth between selected shelves while making the choice and some of them were categorised as confused (BIAC) shoppers. Most shoppers executing the BIAC task spent a lot of time walking around the shop. Amongst the confused category, there were some shoppers who had decided that they would buy a souvenir, while there was another category of shoppers who were confused whether to buy a piece of clothing or a souvenir. Finally, for the NBI category, the unpredictability of the human was evident. Even though we had instructed the shoppers that they had to window shop, we ended up with 6 impulsive shoppers - these shoppers initially started off with normal window shopping, but at some point, they liked an item and considered it like a confused buyer. After the study was completed (turning of the sensing application), these shoppers ended up buying the item. So evidently, some of these impulsive shoppers ended up being in the confused category.

6.5.2.3 Information Gain

Finally, we analyse the information gain of the various features in the study. Figure 6.8 shows the features with the highest information gain in Study 2. From the data we see that Stand and Walk - indicating the count of state transitions to Stand and Walk have the highest information gain. A possible reason for this is that customers who are focused usually tend to stick to one location (also reasoning why stand duration is ranked fifth), while in case of confusion, participants would walk

around, stop and then walk again to a different shelf. This also gave the intuition as to why euclidean and Manhattan distance measures featured higher in the top feature list.

Comparing the features with the highest information gain in the two studies, we find that the type of store plays an important role in determining which features will have the highest information gain and in turn, the highest influence in determining the behavior.

6.5.2.4 Sensitivity Analysis

The number of episodes used in our studies is small and for our studies, we have performed a leave one episode out cross validation. The classifier model generated for every episode in the study is susceptible to overfitting. To understand if this is really an issue, we performed sensitivity studies on various sizes of the training dataset.

For this study, we divided the entire dataset (U) into two sets – the training set and the testing set. The size of the training set (S_{Train}) was varied between 10% of U to 90% of U in steps of 10. A stratified random selection approach was used to create the training set of size S_{Train}. This ensured that the training model has a balanced representation of each class label ('BIAF','BIAC','NBI'). Episodes that were not used in creating the trainer were assigned to the testing set. Once the training set was selected, it was clustered into $k = 3$ clusters and for each cluster, a classification model (decision tree) was created. To test the performance, each episode from the testing set was selected and assigned to a cluster (using k-means clustering approach). Inside the cluster, the episode was passed through the cluster specific classification model and the behavior exhibited by the shopper in the episode was determined. For every value of S_{Train}, the process was repeated 20 times with unique seeds, to create 20 different unique training sets.

Figure 6.9 represents the performance of *CROSDAC* for both binary as well as 3-ary classification. The error bars represents the standard deviation in accuracy for

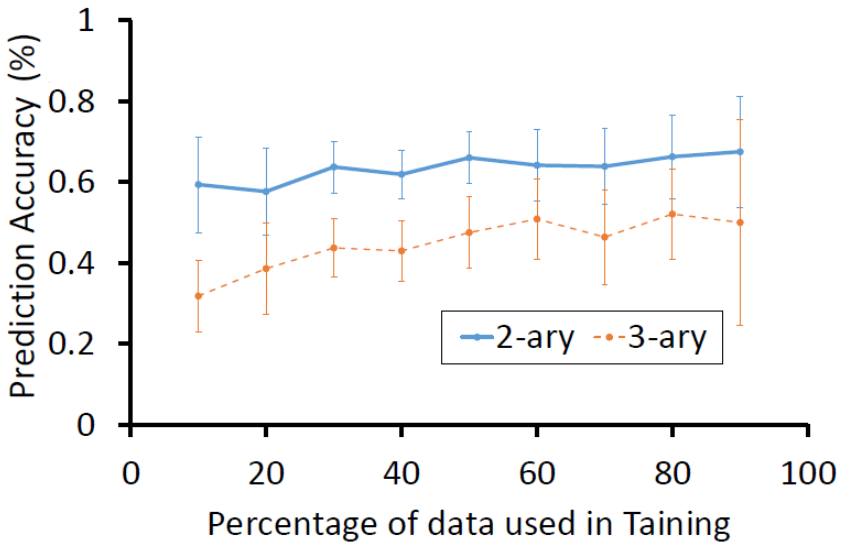


Figure 6.9: Average Performance (20-Runs) of *CROSDAC* for Different Training Data Size

the 20 runs. From the figure we can see that the overall variance in performance of the binary classifier – distinguishing between 'BI' and 'NBI' was lower than the 3-ary classification. The overall performance accuracy for the 2-ary classification is between 60% and 70%. Compared to this, the 3-ary classifier accuracy varies between 32% to 51%, where 32% accuracy was achieved for $S_{Train} = 10\%$. For 3-ary classification, random selection probability is 33%. This indicates that when the training dataset size is small, the performance is similar to random selection. However, as more data is used to train the model, the performance gradually improves. The difference between the system's performance when $S_{Train} = 50\%$ and $S_{Train} = 90\%$ is used is 3% as compared to a difference of 13% between $S_{Train} = 10\%$ and $S_{Train} = 50\%$, indicating that when the training set size is small, there is variation of performance from 3-ary, which gradually stabilises as reasonable amount of training data is available. In case of binary classification, the increase in performance of the technique is marginal for various S_{Train} (there is a difference of 1.5% between $S_{Train} = 50\%$ and $S_{Train} = 90\%$) indicating that the technique's behavior is consistent – increase in the training data size does not significantly improve the performance. Since in these experiments, we have varied the training set

for every run and have also varied the size of the training set, yet we achieved similar performance for the different runs, indicating that the model is not overfitting.

6.6 Discussion

In this Chapter, I have shown two studies that were conducted to determine the shopping behavior. There are some interesting points of discussion, which I highlight in this section.

Location Specific Model: From the two studies that we conducted, we found that there is ample difference between the behavior exhibited by shoppers in a food court as compared to the souvenir store and that is expected. This difference is also highlighted by the ranking of features in terms of information gain. There can be many other behaviors - e.g. behavior in a shoe store or a supermarket will be different. However, for every location, there are certain fixed traits exhibited which can determine the intent of the shopper - e.g. a confused shopper might try out multiple sunglasses or in a food court, a confused shopper might stare at the menu longer. So we should build classification model for a set of similar stores.

Location specific Prediction Approach: For our current studies we used indoor localization techniques which are known to have atleast 2 meters inaccuracy. In case of the foodcourt, the length of the floor was more than 30 meters and the average distance between cash registers of stores was more than 2 meters and so a 2 meter inaccuracy was tolerable. However, in case of the souvenir store, many racks were within 1 meter of each other and thus indoor location techniques have errors in determining the exact shopper's location, which in turn propagates error to the semantic level features. Thus, location specific techniques to mine out the micro features should be used – e.g. techniques such as BLE localization might help in improving the localization errors in a small store as compared to the Wi-Fi based location prediction techniques.

Alternate Sensors: In our current studies we have attempted to determine the be-

havior using the data from the smartphone. However, with the prevalence of other sensing options - smartwatches or fitness bands, it might be possible to extract a richer set of features - e.g. currently we have information about where a person is standing and for how long - smartwatches can enhance this feature by adding how many items were picked while standing at the location to the current feature set.

Handling the Cold Start: Since the system is designed to determine the behavior of the shopper unobtrusively and without any personalised training, getting the initial corpus of diverse sensed data might be difficult. Even with data from a small set of volunteers, it might not be possible to cover a wide range of shopping styles. A possible approach to handle such a situation is to involve the user - e.g. if the prediction probability of a certain behavior is low, then the application might ask certain questions to the user and determine the behavior, which can be added to the database. Alternately, instead of employing a deterministic behavior prediction system, future systems can have probabilistic predictions. In such cases, shopkeepers can take decision if the probability of a certain behavior is above a threshold.

Energy Consumption: Currently for this work, we do not consider energy factor. However, if all the sensors are turned on continuously, battery drain will be high and the device will not be usable for its regular usage like making calls. Thus, a system like *SHOP* should be built while ensuring that the battery drain of the system is not high. Various techniques exist in literature - e.g. duty cycling, adaptive sensing etc. which can be used to conserve the energy.

Possible Alternate Approaches: In this chapter we described a system which utilises sensor data from the smartphone to determine a shopper's behavior. We also discussed about the possibility of using the wearables for such inference. However, in an eco-system without personal devices, alternate possible approaches could be - (a) video analytics - continuous image processing on frames extracted from a video could help in identifying shopper behavior, (b) infrastructure sensing - recent work like ShopMiner [135] have shown that information from infrastructure sensors (RFiD tags) could be used to determine items picked and correlation between items.

Extracting information - e.g. how many times was the same item picked, could help in identifying behavior. However, Shopminer does not create an individual specific item information, thus designing techniques to relate an item interaction with a person is required.

6.7 Summary

This chapter tackles the important, albeit ambitious, problem of inferring the abstract intentions of individual shoppers at stores, based on mobile sensing of their in-store physical behavior which can reveal various interesting insights about the shopper's behavior. The key goal is to develop an activity/intent recognition algorithm that works at crowd-scale in real-life, i.e., it accommodates the diversity expected across the throngs of shoppers, but does not resort to building individualized supervised classification models (which require infeasible amounts of training data). Based on two real-world studies, it was observed that the impact of diversity on certain high level activities, such as shopping, cannot be factored in simply through individual demographic/environmental components. Instead, *CROSDAC* approach, which utilizes an unsupervised clustering algorithm to detect the latent shopping styles embedded in a crowd-scale population, performs better than prior community-oriented approaches, that assume that similarity in demographic attributes translates to a similarity in behavioral styles. *CROSDAC* achieved reasonable accuracies in determining behavior in both our studies, even when noisy sensor data in a real-world setting was used. Although our overall accuracy in either setting was not very high, our studies do indicate that locomotive and trajectory-based features can reveal insights into a shopper's mindset, especially if we employ unsupervised clustering to first disaggregate users into distinct shopping styles. Moreover, the classification accuracy can be expected to increase as more accurate and diverse sensing techniques (e.g., finer-grained BLE based indoor localization, wearable-sensor based gesture monitoring) are adopted. As wearable devices become more

popular, it can be expected that the range of physical activities captured will only increase, thereby bringing more discriminative power in determining much finer grained behaviors.

Chapter 7

Discussion and Future Directions

In this dissertation, I described several approaches for building systems/designing techniques to monitor various daily life activities through multi-modal sensing. In the process, I have addressed various system related challenges - e.g. power, latency etc. To validate these systems, I have conducted several user studies, more than what has been described in this dissertation. Some studies have not been reported as no useful outcome was derived from those studies. Even though useful results were not derived, yet those studies helped in improving all the current systems. Studies described in this dissertation involved 116 participants, who participated in multiple controlled, semi controlled and in-the-wild studies. These studies have resulted in the multiple published [120, 130, 133] or under review works. The food journaling application – *Annapurna* has been demonstrated in various conferences and seminars (Mobisys 2016 [134], ICDCN 2016 [1], TechInnovation 2016 [2]).

During this journey, I have learnt multiple important lessons, including:

- A major lesson that I learnt while testing the systems is not to depend on models created using data collected from controlled studies for determining real world gestures. I found that a user behavior in a controlled study was very different from a real world study. There are numerous scenarios and environ-

[1] http://ares.smu.edu.sg/icdcn16/posters.html
[2] http://www.techinnovation.com.sg/

ments that a user faces, which a lab study cannot predict – e.g. eating in a food court with a group of friends usually results in more random hand gestures as compared to an in-lab data collection. In short, models created in a lab setting more often than not, fail in the real world, even if cross validated accuracy of such models are high. Thus, anyone interested in building systems to monitor real-world activities, should ensure that they have collected reasonably large amount of data from real-world settings. It is not necessary that the data has to be personalised, but it should capture a diverse activity set which will be representational of activities that the user of the system might perform.

- Another key takeaway from the system building was that in terms of energy consumption of the system, no component inside a smartphone or any wearable is a cheap component and no sensor is a cheap sensor. Turning on any sensor, no matter how cheap it is, will affect the overall system's lifetime and in turn performance. So, systems should be built while balancing the duration for which a sensor is operational and the accuracy desired for the system.

- Finally, for conducting user studies, unless the entire process is performed systematically – from planning to post processing, reworks becomes unavoidable. An example of such a situation arose during an initial data collection for the I^4S study. During the data collection phase, shadowing the shopper was done by two of us. After the first round of data collection, we identified difference in the shadowing approach that each of us had taken. While I had marked pick when the hand touched the item, my colleague had marked pick after the hand had removed the item from the shelf. So, in both the cases, even though individually both the markings were acceptable, however care had to be taken during the processing. This by itself was not a show stopper. However, when this was added to the next difference that we had, data processing became more difficult. The next difference was the orientation of the hand at the start of the shopping – we learnt that the performance of the game

rotation vector sensor data was affected by the initial orientation and since we had collected data without actually ensuring similar hand orientation, or noting down the initial hand orientation, some of the initial data collected could not be analysed and used in our studies.

These experiences in previous projects will definitely help in better executing future research. I next describe some future research possibilities.

In this dissertation, I have presented some novel and innovative approaches and techniques that I have applied to produce energy-efficient, accurate and non-personalised systems for ADL monitoring. During the process of building these systems, various innovative research directions have opened up, which goes beyond the food journaling or shopping activity/intent detection. In this section, I discuss some possible improvement to the existing systems and applicability of the currently developed approaches in other activity recognition systems, as well as identify additional interesting research questions for future work.

7.1 Additional Uses of Gesture-Triggered Image Capturing

The *Annapurna*-like approach of gesture-triggered image capture by the smartwatch need not be restricted to eating, but can be used to capture context of other activities such as shopping. In particular, we explored this concept for identifying the items with which a shopper interacted in a store – i.e., in achieving the same goals as I^4S, but without the infrastructural BLE support.

7.1.1 In-Store Interaction Monitoring

Chapter 5 describes I^4S, a system that fuses sensor data from multiple source, to identify the location from where an item is picked. Through a simple lookup, I^4S can identify the exact item picked (not currently implemented). In addition to I^4S,

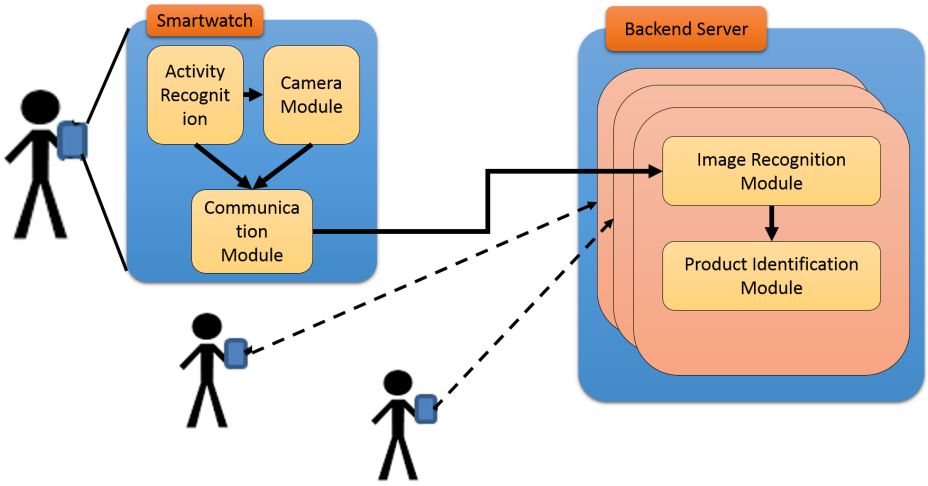


Figure 7.1: Architecture for a Single Device Item Identification System

we have also explored a single device in-store item interaction technique, which is similar to *Annapurna*. Through a controlled study, we found that an *Annapurna*–like approach could be used for in-store item interaction. The approach, similar to I^4S, utilises the inertial sensor data information from the smartwatch to determine the picking gesture. But instead of using the BLE scan information, this approach investigated the use of a camera (attached to the smartwatch) to determine the object being picked - similar to *Annapurna*. However there are a few differences between this technique and *Annapurna*– (a) since the picking gesture involves hand movement in a particular direction, the camera orientation that was used in *Annapurna* might not be optimum, (b) *Annapurna* did not attempt to identify the food item from the image and (c) Unlike picking, eating is a repetitive and periodic gesture. We thus had to continuously identify the hand gesture. In this work, we did not target developing a real time solution and hence I will not discuss (c), but focus only on the first two differences. At a high level, this approach uses the inertial sensor of the smartwatch to determine "pick" gesture and captures images using the camera on the smartwatch to identify the item. The *identification of pick* in this approach is similar to I^4S and we will not focus on that. The subsequent sections focus only on the image capturing and identification technique involved.

7.1.1.1 System Overview

Unlike I^4S, where we identify the location of pick and then through a reverse lookup we identify the item picked, the goal of this approach is to directly identify the item picked through the images captured. Thus, the whole working of the system can be broken down into two parts: (a) identify the pick gesture and (b) capture images and identify objects in the image. In this section we concentrate only on (b).

To realise (b), consecutive image frames captured by the smartwatch's camera are transferred to a backend server. The image processing module on the server matches the image frame captured by the watch against a corpus of test images, to determine the object in the image.

Figure 7.1 shows the overall working of such a system. In this architecture, the smartwatch is responsible for "pick" detection and "image capturing", while the backend server is responsible for the item identification through image recognition.

7.1.1.2 Design Choices

To realise the system, we required a watch which could capture image when a person was picking. We tested out various camera positions on the watch's strap as well as the orientation of the watch with respect to the hand. Based on our empirical observations, we found that we could capture the best images when the camera was on the side of the watch face and the watch was rotated so that the face was on the same plane as the person's palm. Based on this requirement, we found that the Omate TrueSmart [100] smartwatch had a camera on the watch which best suited our needs. Figure 7.2 shows the orientation of the watch as well as the camera position on the watch. It also shows the image captured by the watch in a frame, while Figure 7.3 shows the images captured by the smartwatch. The time difference between two successive images in the figure is 167 ms.

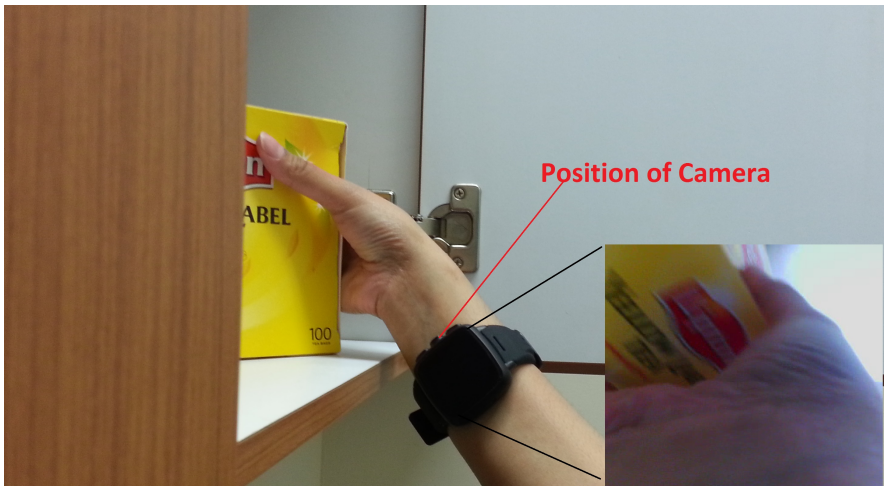


Figure 7.2: Camera View With Image Captured While an Item is Being Picked

7.1.1.3 Dataset

To understand the feasibility of capturing the item's image, we performed a small lab study. For our study, we recruited 5 participants from our lab (2 males, 3 females - all aged between 20 and 30) and who were almost in the similar height range (1.55 to 1.70 meters). Participants were asked to perform activity sequences that are normally carried out while grocery shopping: (1) open the door of the shelf, (2) pick an item from the shelf, (3) put the item aside and (4) close the door of the shelf. Each participant repeated the sequence for 3 different items (packs of biscuit, packs of green tea and bottles of water) which were placed in 3 different shelves. Each participant repeated this process 10 times. In total, we collected 30 sample sequences each from a participant. The ground truth of the activities was collected by a shadower, who labeling the activities as the user performs it.

7.1.1.4 Methodology

As shown in Figure 7.3, while performing our experiments of picking items, we found that for a short period of time, the camera on the watch usually points towards the item that is being picked and it is possible to capture a legible image of the item being picked from the camera. To investigate the possibility of identifying the item while a person is picking and to identify the best moment when the object is

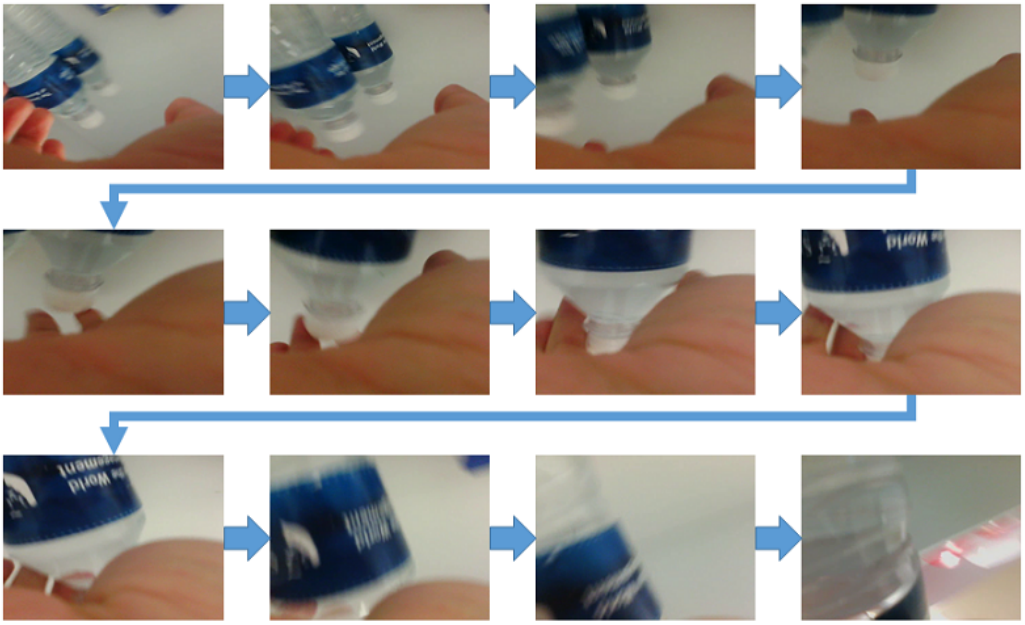


Figure 7.3: Images Extracted from the Video While Person is Picking an Item

visible, we captured a video during the participant's activity sequence and extracted all individual frames from the video. This was done for the 5 users, each picking items from the three different racks.

In all we extracted frames from all the 150 videos that were captured. Next, to analyse if the captured image (extracted from the video frame) could be identified automatically by a image recognition software, we used the as-is implementation of SURF [12] algorithm in openCV. A small training set was created by capturing the images of the 3 objects (3 images of each object, taken from 3 different angles). Each frame was compared against the training set and the recognition was considered successful if the image was identified correctly.

7.1.1.5 Results

We first investigate if all the frame extracted from the video captured the image of the item. Based on manually inspecting the 150 videos, we found that we could see the object at least once in all the 150 videos. Next, from all the videos, frames were extracted. We again manually labeled whether the frame captured the image of the item. We plotted the probability of a captured image being 'useful' (i.e.,

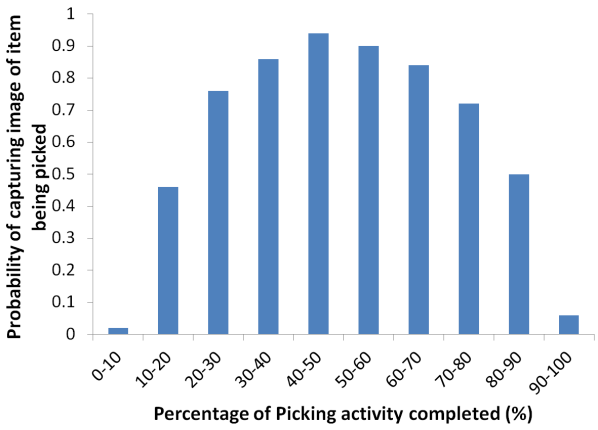


Figure 7.4: Probability of Capturing Image of Item Being Picked

provides a clear view of the item picked) as a function of the time when the image was captured, relative to the overall duration of the gesture. Figure 7.4 shows the plot of this probability as a function of the time, with the time being expressed as a percentage of the overall gesture duration. We can see that the probability of getting a useful image is within the 20^{th} to 80^{th} percent of the duration. In terms of absolute time, this window is approximately 1.2 seconds, which is a fairly wide window and thus instead of capturing a video, even if a single image is captured, the image of the item will be procured, resulting in savings in the overall energy consumption.

We next analyse the performance of the SURF algorithm in identifying the item in the images. We found that we could identify the correct item in 61% cases (vanilla baseline - 33%), which was not very high. When we analysed the images that were identified wrongly, we realized that the wrong classification occurred due to (i) occlusion of the object - if the object is small, the fingers cover a major portion of the image. In our case, we found that in many of the frames containing the pack of biscuit, where part of the packet can be seen, the image recognition software mistook it for the tea box. (ii) blurriness - when the item being picked is picked, motion blur creeks in the image frame obtained from the video. This might result in mis-classification. (iii) Insufficient training of the recognition model - for our small study, we just used simple feature matching to recognise the objects. Even though these images were taken from different angles, they did not cover all possible angles

that might be visible to the camera when the object is being picked. The accuracy should improve with a more sophisticated recognition model that can be trained carefully for this application. To understand if images captured by the watch were identifiable by a commercially available deep learning based image recognition software, we ran some of the images on Clarifai [27], a commercial image recognition software which uses convolutional neural networks. We found that, even without supplying training data, the deep learning software was able to broadly tag the images obtained from our study and every video had at least 1 frame which had been identified correctly by Clarifai.

7.1.1.6 Comparing with I^4S

Both I^4S and the *Annapurna*-like approach have their own advantages and disadvantages. We next compare the two approaches for some important aspects:

Privacy: Compromise of privacy is a growing concern for many wearable applications. However, since privacy concerns is person dependant, it is difficult to determine which sensors data leakage has more serious impact. For example, a person X might be okay with automatic camera trigger, but might not want her location information to be public, while another user might be okay with location leaks, but will be concerned about her accompanier details getting leaked. I compare the two approaches in terms of location privacy and image captured concern. In terms of location leak - identifying that the shopper is in a particular shop or has picked a particular item is possible through both the techniques. However, privacy concerns such as capturing images of all other shoppers who are present in the shop at a particular time is possible in the *Annapurna*-like approach. Unless proper precaution is taken, privacy can be a major concern in the *Annapurna*-like approach. Approaches such as on-the-fly face detection and blurring can be applied, but that might not be adequate in many scenarios.

Occlusion: Since the *Annapurna*-like approach utilises the camera mounted on the smartwatch to capture images of the products, it requires the smartwatch

camera to have an unobstructed view i.e., not be covered by clothing such as jacket or shirt sleeves. An alternative to the use of wrist worn cameras can be the use of smartglasses [122], however this will increase the number of devices involved in the recognition. Since I^4S does not capture images, this is not a concern in the approach. However, I^4S uses RF signals to determine the precise location of the person and the signals are affected when there is an obstruction between the beacon and the device.

Energy Overhead: Currently both the approaches have a certain set of sensors which are continuously sensing. For I^4S, the sensors includes the inertial sensors and continuous low energy bluetooth scan, while for the *Annapurna*-like approach, the sensors that are continuously sensing are inertial and the camera. Thus the comparison between the two approaches is the camera versus BLE scans. From empirical evaluation, we found that continuous video recording using the smartwatch drains out a completely charged smartwatch in 80 minutes, while the battery drop for a completely charged smartwatch performing BLE scans is 18% in 80 minutes. This indicates that I^4S has lower energy overhead as compared to the *Annapurna*-like approach. However, smart triggering of the camera (an approach is described in Chapter 4) instead of continuous video recording can significantly lower the energy consumption.

Identifying misplaced items: I^4S*'s* operation is based on the premise that identifying, at shelf-level granularity, the location of a user's pick gesture implicitly identifies the product (or product category) selected. While this is likely to be broadly true, store operators know only too well that products are continually being misplaced by shoppers. Hence, if a shopper picks up an item from a shelf where it has been dumped by a previous shopper, I^4S will result in a mis-identification of a shopper's true interest. On a contrary, the *Annapurna*-like approach is based purely on image recognition and thus immune to item misplacement.

Infrastructure / Store Knowledge Overhead: Since both the approaches requires item level information - a detailed inventory list is necessary for both the

approaches. Besides the inventory list, I^4S requires (a) the knowledge of the shelf location of every item, and (b) deployment of BLE beacons, with the system being aware of the location of the deployed beacon. This is an additional overhead for the system. Alternately, the *Annapurna*-like approach has the additional overhead of maintaining a large corpus of images of the items in the store against which the captured image can be matched.

Multi device synchronisation: I^4S relies on fusion of data from multiple devices – smartwatch, smartphone as well as deployed beacons. Failure due to software or hardware at any source or synchronisation mismatch between any pair of devices will produce erroneous predictions. Contrary to this, the *Annapurna*-like approach relies on a single device and thus is not affected by failure of other devices.

7.2 Short Term Plan

Before discussing my longer term research plans related to the broad topic of ADL monitoring, I discuss some short term plans which can be studied to improve the existing diet monitoring and shopping monitoring systems.

7.2.1 Automated Food Journaling

Energy: The existing *Annapurna* application has been tested with users in the real world setting and as mentioned in Section 4.6, I found that the battery life of the system was less than half a day, indicating that if we had to take the system beyond lab studies, we will have to figure out techniques to improve on the energy. I have listed down some possible energy improving approaches in Section 4.6. I plan to test those techniques and analyse the possibility of increasing the system life.

Hardware Independence: The current version of *Annapurna* requires a camera on the smartwatch, that too at a certain position. However, due to various concerns (energy, privacy, lack of compelling use case), manufacturers are gradually removing the camera from the smartwatch and this will affect the existing design.

To achieve an image based journal, alternate devices – smartglasses or other head mounted cameras might be used, with gesture based triggering used to capture the images. Alternately, if the system can seamlessly integrate with infrastructure cameras, they might be useful in capturing images.

7.2.2 In-store Interaction Identification

Using video cameras: Both these approaches have their own advantages as well as disadvantages. Other than these approaches, there are various other possibilities that can be explored to identify in-store item interactions. Some other possibilities includes - combining infrastructure-based video sensing with either approaches to improve the accuracy of pick identification and localization. For example, video cameras mounted on either walls or on the top of individual racks may be used to identify the time instants when a shopper's hand picks up an item from a shelf, and this time may be correlated with the inertial sensing-based pick time detected by the smartwatch to unambiguously identify which shopper picked up the specific product.

Futuristic Shopping Experience: In the current work, I have concentrated mainly on the picking gestures. However, there are many other gestures that are performed in the store. An interesting direction can be in creating a taxonomy of all possible gestures, which can be used to identify state transition probabilities, which can explain shopping event sequence. Currently, in this dissertation I have not looked at any real time prediction techniques. With the knowledge of state transitions, various existing techniques can be used to determine the shopper's behavior - e.g. if a shopper picks item A and then stands for a while, it can be predicted that he will pick item B. With techniques like I^4S in place, various innovative in-store experiences can also be created for customers – e.g. a smart basket along with the I^4S system can help in identifying all items that have been placed in it. When the smart-basket $+ I^4S$ system detects that the shopper has finished picking all items, it automatically

checks out all the items.

Beyond the Shops: Techniques learnt and used in building the above systems can be used to build other innovative applications which can involve usage of one or more devices amongst a smartphone, wearable device like a smartwatch and any infrastructure sensors. Some examples can be: With the availability of smart displays in public places [142], a possible use case of a system like I^4S can be in the library. When a person picks a book from the library's shelf and reads the synopsis, the smart display might communicate with the smartwatch and identify the book picked. With that information, the display can show details of similar books or reviews from other users etc.

7.3 Longer Term Research

Throughout this dissertation, I have not only described the systems that we have built, but I also shared the experiences gathered while building the systems. I believe that the lessons learnt while building these systems will have a much deeper impact on future systems that will use mobile, wearable and infrastructure sensor data to monitor lifestyle. The systems described in this dissertation monitor a few example daily life activities. These systems have been tested on a small group of participants. There are several possible extensions to the approaches and techniques:

Possible Extended Use-Cases: This dissertation describes some possible approaches to monitor two common daily lifestyle activities (eating and shopping). Researchers or system developers can extend these approaches to monitor various other daily life activities. For example, a system similar to I^4S can assist in monitoring the cooking activity. Such a system can utilise the sensor data from wearables and infrastructure sensors. Monitoring the cooking activity can help in identifying whether all ingredients have been correctly added to the food item that is being prepared. Additionally, it can also determine if the quantity of the added ingredients

are correct. Such a system is useful in any environment where cooking takes place – be it the kitchen of a house to the kitchen of a restaurant. The system will require inertial sensor data from the smartwatch to determine when an ingredient is added. It will require the indoor localization (which might be BLE beacons installed on racks/shelves) to determine the location from where an ingredient is picked.

Other than general daily life activity monitoring, I believe that techniques described in this dissertation will be useful in the elder care domain. For the elder care domain, a smartwatch based solution can be useful in determining if an elderly individual has performed one or more daily life task – e.g., if she has consumed her medication. For the medicine intake monitoring example, inertial sensor data from a smartwatch (or any hand worn device with sensing capabilities) can be used to determine the *taking medicine gesture*, which might involve steps similar to (a) opening the medicine box, (b) identifying the medicine strip, (c) taking out the capsule from the strip, and (d) consume the capsule. For such a solution, the inertial sensor can identify gestures such as opening box or putting the medicine in mouth, while the camera can be used to capture the image of the strip of medicine from which the capsule was extracted. This system is similar to the *Annapurna* system that has been described in this dissertation.

Impact of Users: The systems described in this dissertation has not been tested on diverse user demographics. While the I^4S system has been tested on students in the university, *Annapurna* has been tested on members of our lab. Although these studies successfully demonstrated the proof of concept, there might be other factors to consider while expanding to other demographic groups. As mentioned previously, one demographic group which I am particularly interested in and I believe will benefit from automated daily life monitoring is the elderly. Automated and unobtrusive monitoring of their daily life activities can help in identifying and improving the assistance that they require. However, there are several challenges in monitoring the elderly, one amongst which is the reluctance of the elderly to use wearables [35]. Thus, if ADL monitoring techniques for this category have to be

designed and it has to be performed using wearable devices, innovative approaches have to be taken - e.g. using wearable rings, which hypothetically, might be less noticeable to the end user. Alternately, innovative use of infrastructure sensors can be used for the activity monitoring. An interesting direction of research could be in determining how an existing system can be modified so that it can cater to a specific demographic group. Currently, I have not tested any of the systems on the elderly and thus I believe that performing studies on them will help in identifying challenges specific to the demographics.

Impact on Users: Currently, I have built systems which can help in identifying ADLs. The next obvious question is: *what do we do with these ADLs?* I believe that an interesting future research direction will be in understanding user needs. For example, in case we identify eating, what useful analytics should we provide to the users? Based on a survey, we found that most respondents wanted *Annapurna* to determine the number of calories consumed in a meal. However, with existing techniques, determining the number of calories is a hard problem. So an interesting research direction is in identifying the sweet-spot between what a user wants and what can be provided to the user.

ADL specific features: Currently in *CROSDAC*, I have used features which are specific to the activity. However, most existing ADL monitoring applications use a set of statistical features - mean, variance, correlation etc. I believe that even though the statistical features have been powerful in micro activity recognition, to identify an ADL, a more sophisticated set of features are needed. I believe that identifying and analysing these features will be a useful direction in ADL monitoring. Alternately, recent work in deep-learning based activity recognition methods illustrates a direction, where feature-less ADL monitoring might be possible.

Evaluation of Annotation Techniques: To ensure accurate marking of ground truth, each and every ADL step should be recorded and an annotator should be able to look back at the recording and mark the ground truth. However, this is not possible because (i) looking back at a recording again and again is not a scalable

solution, (ii) IRB committees will have privacy concerns and (iii) the labeling still will be done by a human.

Currently in my studies, I have used a manual shadowing approach and assume that the ground truth marked is 100% accurate. But since there is human involvement, there will definitely be errors in the ground truth marking. Thus, an interesting research direction is to analyse the reliability of ground truth annotations in ADL monitoring applications. To do this, one approach can be in having multiple people marking the same ADL episode and then analysing the variance in the ground truth marking. Based on findings from this analysis, ground truth annotation correction techniques can be determined.

Classification Techniques: Currently, I have only evaluated shallow learning classification techniques. However, with the rapid increase in the number of devices and sensing capability of these devices, in future, shallow learning approaches will be laborious. With the evolution of deep learning and its success in certain domains, researchers have started exploring deep learning techniques for activity recognition [17, 61]. The advantage of deep learning over shallow learning is that shallow learning activity recognition classifiers require a set of features. For ADLs, the feature set should accommodate the ADL specific feature (e.g. [104, 130]), indicating that it should be hand-crafted, which by itself is a challenging task. Since deep learning does not need features, this hand-crafted feature identification step is not involved in deep learning.

With the rapidly increasing number of devices, deep learning (focusing on representational learning) also seems well suited for transfer learning (one device training the other) as compared to shallow learning. Since each device will produce a certain sensor stream, which might not be similar to one another, work such as [14] have shown that deep learning performs better than shallow learning even when the training and test sets are not similar. Thus an interesting direction will be to evaluate the performance of deep learning for multi-modal ADL monitoring.

Chapter 8

Conclusion

With the continuously increasing number of sensors in our personal devices as well as in the environment around us, monitoring basic activities and context has become a reality. This has opened the floodgates for fine-grained, multi-modal monitoring of more complex activities, which in turn can provide useful details and insights – e.g. remembering what you ate two days ago for lunch will become easier.

This dissertation has shown that *it is indeed possible to harness the multi-modal sensing capabilities of commercial, off-the-shelf mobile, wearable and IoT devices to derive accurate and fine-grained insights about multiple different aspects of an individual's daily lifestyle activities and behavior specifically related to retail shopping and eating.*

This dissertation, describes various techniques and approaches for building these daily life activity monitoring systems. It also shows the possibility of identifying the in-activity user behavior. The next section provides a quick recap of these systems and techniques described in this dissertation.

8.1 System and Technique Summary

Annapurna: In Chapter 4, I have described *Annapurna*, an automated diet monitoring and food journaling application. *Annapurna* has a smartwatch module, a

smartphone module and a server module. *Annapurna* primarily relies on a smartwatch to determine eating gesture and capturing images of the food items being consumed. The working of *Annapurna's* smartwatch module is divided into multiple layers and various design choices have been made at every layer to either save energy or increase the probability of capturing an image. Some energy saving techniques that have been taken for the watch module included - multi leveled inertial sensing and camera triggering, capturing images from preview mode and storing lower quality jpeg files.

Images captured by the *Annapurna's* smartwatch module were transferred to the smartphone module, which performed initial image filtering and passed the relevant images to the server. The server applied multiple heuristics to determine whether the food plate was visible in any of the images and which were the best images in terms of capturing the food plate.

Through multiple real-world user studies (7 users over 12 days), I showed that *Annapurna* has minimal false positive and false negative rates of 6.5% and 3.3%, respectively, while recognizing a wide variety of food items, consumed at various locations, by people of different nationalities, and with different eating styles. An initial version of this work, showing feasibility of the technique has appeared in a PerCom 2015 workshop [133], while the experience with developing a robust gesture recognizer has been accepted in the WPA workshop [132]. An article detailing the image capturing strategy is currently under submission.

I*^{4}*S: I^4S, a system to automatically identify items that a shopper interacted with, has been described in Chapter 5. I^4S utilises sensor data from the smartwatch, the smartphone as well as information from BLE beacons to determine the interacted item. While both the smartwatch and smartphone continuously captures the inertial sensor data, the smartphone additionally captures BLE scan information. Using gesture recognition techniques, the smartwatch determines if the shopper is interacting with an item. When an interaction is determined, the BLE scan information assists the system in identifies the shopper's rack level location. In addition to the

gesture recognition, the smartwatch is also used to determine the shelf as well as zone within the shelf from where the item was picked.

Through an comprehensive user study of 31 shoppers, I have shown that we could identify various aspects of the item interaction - identifying picking gesture, identifying location of pick and identifying the sub-shelf level picking location accurately in over 85% cases individually, indicating that identifying item interaction is indeed possible. This work is currently under submission.

In Chapter 7, I have also described an alternate technique to identify items that is picked by a shopper. In this technique, we used the camera in a smartwatch to determine the picking gesture and triggered the camera appropriately to capture the image of the item picked. Through a small user study, we demonstrated that it was possible to capture the image of the item being picked. This work was published in COMSNETS 2016 workshop [120].

CROSDAC: Chapter 6 shows a technique of using sensor data from the shopper's mobile device to determine the shopping behavior and intent – whether the shopper has buying intention or not and if she has buying intentions, whether she is focused or confused. *CROSDAC* demonstrates that the accuracy in determining shopping behavior increases when shopper are clustered based on shopping style and then their behavior is determined. Through two studies conducted in diverse settings (study with 30 users in a food court a shopping mall in New Delhi and with 22 users in the gift shop of our University in Singapore), I showed that *CROSDAC* approach performed better than alternate existing approaches. This work has appeared in the Proceedings of International Symposium of Wearable Computing 2015 [130] and is being readied for a journal submission.

8.2 Closing Remarks

This dissertation demonstrates that daily lifestyle monitoring through fusion of sensor data from multiple sensors located in either one or several devices can be done

reliably and accurately using various off-the-shelf devices. This is a paradigm shift from wiring sensors on an individual with the intention of monitoring basic activities to using personal devices to unobtrusively monitor an individual. This possibility of unobtrusively monitoring an individual will not only assist in building applications for healthy living, but will also open up the floodgates for various innovative applications in various other domains, some of which I have mentioned in the motivating scenarios throughout the dissertation.

In this dissertation, through detailed evaluation of multiple techniques (either with single device or multiple devices), I have demonstrated the possibility of monitoring common daily life activities. I have discussed various system level challenges that have to be answered so that these techniques can cross over from being a proof of concept to an actual usable system. I believe that techniques shown in this dissertation will help drive innovative daily life monitoring applications which can be used by an entire population.

This dissertation has also demonstrated that it might be possible to identify user's behavior through the analysis of sensor data from the user's personal devices. Currently, this is in a developing stage and through further improvement, these techniques can not only assist in monitor ADLs, but also predict the intent of the user, which in turn will enable pre-emptive intervention where necessary.

Collectively, the ADL monitoring and behavior determination techniques discussed in this dissertation paves the way for future daily life monitoring applications. These applications can run on multiple off-the-shelf, utilising several sensor classes to provide fine-grained details of activities to the individual.

Bibliography

[1] F. Adib, Z. Kabelac, D. Katabi, and R. C. Miller. 3d tracking via body radio reflections. In *NSDI*, volume 14, pages 317–329, 2014.

[2] Amazon go. `https://www.amazon.com/b?node=16008589011`. Online; Accessed: 2017-03-29.

[3] O. Amft, H. Junker, and G. Troster. Detection of eating and drinking arm gestures using inertial body-worn sensors. In *Wearable Computers, 2005. Proceedings. Ninth IEEE International Symposium on*, pages 160–163, Oct 2005.

[4] O. Amft, M. Stäger, P. Lukowicz, and G. Tröster. Analysis of chewing sounds for dietary monitoring. In *International Conference on Ubiquitous Computing*, pages 56–72. Springer, 2005.

[5] O. Amft and G. Tröster. Recognition of dietary activity events using on-body sensors. *Artificial Intelligence in Medicine*, 42(2):121–136, 2008.

[6] Android ActivityRecognitionApi. `https://developers.google.com/android/reference/com/google/android/gms/location/ActivityRecognitionApi`. Online; Accessed: 2017-03-20.

[7] W. Applebaum. Studying customer behavior in retail stores. *Journal of Marketing*, 16(2):172–178, 1951.

[8] M. Azizyan, I. Constandache, and R. Roy Choudhury. Surroundsense: Mobile phone localization via ambience fingerprinting. In *Proceedings of the 15th Annual International Conference on Mobile Computing and Networking*, MobiCom '09, pages 261–272, New York, NY, USA, 2009. ACM.

[9] P. Bahl and V. N. Padmanabhan. Radar: An in-building rf-based user location and tracking system. In *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 2, pages 775–784. Ieee, IEEE, 2000.

[10] R. K. Balan, M. Satyanarayanan, S. Y. Park, and T. Okoshi. Tactics-based remote execution for mobile computing. In *Proceedings of the 1st International Conference on Mobile Systems, Applications and Services*, MobiSys '03, pages 273–286, New York, NY, USA, 2003. ACM.

[11] L. Bao and S. S. Intille. Activity recognition from user-annotated acceleration data. In *Pervasive computing*, pages 1–17. Springer, 2004.

[12] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded-up robust features (surf). *Computer vision and image understanding*, 110(3):346–359, 2008.

[13] F. Ben Abdesslem, A. Phillips, and T. Henderson. Less is more: energy-efficient mobile sensing with senseless. In *Proceedings of the 1st ACM workshop on Networking, systems, and applications for mobile handhelds*, pages 61–62. ACM, 2009.

[14] Y. Bengio, F. Bastien, A. Bergeron, N. Boulanger-Lewandowski, T. Breuel, Y. Chherawala, M. Cisse, M. Côté, D. Erhan, J. Eustache, X. Glorot, X. Muller, S. Pannetier Lebeuf, R. Pascanu, S. Rifai, F. Savard, and G. Sicard. Deep learners benefit more from out-of-distribution examples. In *JMLR W&CP: Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics (AISTATS 2011)*, Apr. 2011.

[15] M. Benzeghiba, R. De Mori, O. Deroo, S. Dupont, T. Erbes, D. Jouvet, L. Fissore, P. Laface, A. Mertins, C. Ris, et al. Automatic speech recognition and speech variability: A review. *Speech communication*, 49(10):763–786, 2007.

[16] M. Berchtold, M. Budde, D. Gordon, H. R. Schmidtke, and M. Beigl. Actiserv: Activity recognition service for mobile phones. In *Wearable Computers (ISWC), 2010 International Symposium on*, pages 1–8. IEEE, 2010.

[17] S. Bhattacharya and N. D. Lane. From smart to deep: Robust activity recognition on smartwatches using deep learning. In *2016 IEEE International Conference on Pervasive Computing and Communication Workshops (PerCom Workshops)*, pages 1–6, March 2016.

[18] Y. Bi, M. Lv, C. Song, W. Xu, N. Guan, and W. Yi. Autodietary: A wearable acoustic sensor system for food intake recognition in daily life. *IEEE Sensors Journal*, 16(3):806–816, 2016.

[19] P. H. Bloch and M. L. Richins. Shopping without purchase: An investigation of consumer browsing behavior. *NA-Advances in Consumer Research Volume 10*, 1983.

[20] R. Bodor, B. Jackson, and N. Papanikolopoulos. Vision-based human tracking and activity recognition. In *Proc. of the 11th Mediterranean Conf. on Control and Automation*, volume 1. Citeseer, 2003.

[21] R. R. Burke and A. Leykin. *Identifying the Drivers of Shopper Attention, Engagement, and Purchase*, pages 147–187. Emerald Group Publishing Limited, 2014.

[22] S. Cadavid, M. Abdel-Mottaleb, and A. Helal. Exploiting visual quasi-periodicity for real-time chewing event detection using active appearance models and support vector machines. *Personal and Ubiquitous Computing*, 16(6):729–739, 2012.

[23] J. Chen, A. H. Kam, J. Zhang, N. Liu, and L. Shue. Bathroom activity monitoring based on sound. In *International Conference on Pervasive Computing*, pages 47–61. Springer, 2005.

[24] T. Y.-H. Chen, L. Ravindranath, S. Deng, P. Bahl, and H. Balakrishnan. Glimpse: Continuous, real-time object recognition on mobile devices. In *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems*, pages 155–168, New York, NY, USA, 2015. ACM.

[25] Z. Chen, M. Lin, F. Chen, N. Lane, G. Cardone, R. Wang, T. Li, Y. Chen, T. Choudhury, and A. Campbell. Unobtrusive sleep monitoring using smartphones. In *Pervasive Computing Technologies for Healthcare (PervasiveHealth), 2013 7th International Conference on*, pages 145–152, May 2013.

[26] D. Chu, N. D. Lane, T. T.-T. Lai, C. Pang, X. Meng, Q. Guo, F. Li, and F. Zhao. Balancing energy, latency and accuracy for mobile sensor data classification. In *Proceedings of the 9th ACM Conference on Embedded Networked Sensor Systems*, pages 54–67. ACM, 2011.

[27] Clarifai: Image and video reognition api. `http://www.clarifai.com/`. Online; Accessed: 2017-03-20.

[28] I. Constandache, S. Gaonkar, M. Sayler, R. R. Choudhury, and L. Cox. Enloc: Energy-efficient localization for mobile phones. In *INFOCOM, IEEE*, pages 2716–2720. IEEE, 2009.

[29] E. Cuervo, A. Balasubramanian, D.-k. Cho, A. Wolman, S. Saroiu, R. Chandra, and P. Bahl. Maui: making smartphones last longer with code offload. In *Proceedings of the 8th international conference on Mobile systems, applications, and services*, pages 49–62. ACM, 2010.

[30] L. Deng and D. Yu. Deep learning: methods and applications. *Foundations and Trends in Signal Processing*, 7(3–4):197–387, 2014.

[31] Y. Dong, A. Hoover, J. Scisco, and E. Muth. A new method for measuring meal intake in humans via automated wrist motion tracking. *Applied psychophysiology and biofeedback*, 37(3):205–215, 2012.

[32] Y. Dong, J. Scisco, M. Wilson, E. Muth, and A. Hoover. Detecting periods of eating during free-living by tracking wrist motion. *Biomedical and Health Informatics, IEEE Journal of*, 18(4):1253–1260, 2014.

[33] Euclid analytics. http://euclidanalytics.com. Online; Accessed: 2017-03-27.

[34] R. Faragher and R. Harle. An analysis of the accuracy of bluetooth low energy for indoor positioning applications. In *Proceedings of the 27th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+ 2014)*, 2014.

[35] C. B. Fausset, T. L. Mitzner, C. E. Price, B. D. Jones, B. W. Fain, and W. A. Rogers. Older adults use of and attitudes toward activity monitoring technologies. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 57(1):1683–1687, 2013.

[36] Fitbit. https://www.fitbit.com. Online; Accessed:2017-03-22.

[37] Foodai. https://github.com/foodaiorg/foodai.org. Online; Accessed:2017-03-22.

[38] J. Fricke and C. Unsworth. Time use and importance of instrumental activities of daily living. *Australian Occupational Therapy Journal*, 48(3):118–131, 2001.

[39] J. Ganesh, K. E. Reynolds, and M. G. Luckett. Retail patronage behavior and shopper typologies: a replication and extension using a multi-format, multi-method approach. *Journal of the Academy of Marketing Science*, 35(3):369–381, 2007.

[40] C. Graf. The lawton instrumental activities of daily living scale. *AJN The American Journal of Nursing*, 108(4):52–62, 2008.

[41] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1):10–18, 2009.

[42] J. A. Hartigan and M. A. Wong. Algorithm as 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):100–108, 1979.

[43] B. L. Heitmann and L. Lissner. Dietary underreporting by obese individuals–is it specific or non-specific? *Bmj*, 311(7011):986–989, 1995.

[44] S. Hemminki, P. Nurmi, and S. Tarkoma. Accelerometer-based transportation mode detection on smartphones. In *Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems*, page 13. ACM, 2013.

[45] T. K. Ho. Random decision forests. In *Document Analysis and Recognition, 1995., Proceedings of the Third International Conference on*, volume 1, pages 278–282. IEEE, 1995.

[46] H. Huang and S. Lin. Toothbrushing monitoring using wrist watch. In *Proceedings of the 14th ACM Conference on Embedded Network Sensor Systems CD-ROM*, SenSys '16, pages 202–215, New York, NY, USA, 2016. ACM.

[47] Ibotta. https://ibotta.com/how?st=%23how-app-to-app. Online; Acccessed: 2017-04-27.

[48] idate. http://www.idate.org/en/Digiworld-store/DigiWorld-Yearbook-2015_1009.html. Online; Acccessed: 2017-03-27.

[49] C. Intanagonwiwat, R. Govindan, and D. Estrin. Directed diffusion: A scalable and robust communication paradigm for sensor networks. In *Proceedings of the 6th annual international conference on Mobile computing and networking*, pages 56–67. ACM, 2000.

[50] S. V. Ioannou, A. T. Raouzaiou, V. A. Tzouvaras, T. P. Mailis, K. C. Karpouzis, and S. D. Kollias. Emotion recognition through facial expression analysis based on a neurofuzzy network. *Neural Networks*, 18(4):423 – 435, 2005. Emotion and Brain.

[51] G. Kahl, L. Spassova, J. Schöning, S. Gehring, and A. Krüger. Irl smartcart-a user-adaptive context-aware interface for shopping assistance. In *Proceedings of the 16th international conference on Intelligent user interfaces*, pages 359–362. ACM, 2011.

[52] Y. Kawano and K. Yanai. Foodcam: A real-time food recognition system on a smartphone. *Multimedia Tools and Applications*, 74(14):5263–5287, 2015.

[53] A. J. Khan, V. Ranjan, T.-T. Luong, R. Balan, and A. Misra. Experiences with performance tradeoffs in practical, continuous indoor localization. In *2013 IEEE 14th International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM)*, pages 1–9, June 2013.

[54] D. H. Kim, Y. Kim, D. Estrin, and M. B. Srivastava. Sensloc: sensing everyday places and paths using less energy. In *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, pages 43–56. ACM, 2010.

[55] K. H. Kim, S. W. Bang, and S. R. Kim. Emotion recognition system using short-term monitoring of physiological signals. *Medical and biological engineering and computing*, 42(3):419–427, 2004.

[56] M. Kose, O. D. Incel, and C. Ersoy. Online human activity recognition on smart phones. In *Workshop on Mobile Sensing: From Smartphones and Wearables to Big Data*, volume 16, 2012.

[57] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, Boston, USA, 2012. MIT Press.

[58] J. Krockel and F. Bodendorf. Intelligent processing of video streams for visual customer behavior analysis. In *Proc. of ICONS'12*, 2012.

[59] J. R. Kwapisz, G. M. Weiss, and S. A. Moore. Activity recognition using cell phone accelerometers. *ACM SigKDD Explorations Newsletter*, 12(2):74–82, 2011.

[60] N. D. Lane, S. Bhattacharya, P. Georgiev, C. Forlivesi, and F. Kawsar. An early resource characterization of deep learning on wearables, smartphones and internet-of-things devices. In *Proceedings of the 2015 International Workshop on Internet of Things towards Applications*, pages 7–12, New York, NY, USA, 2015. ACM.

[61] N. D. Lane and P. Georgiev. Can deep learning revolutionize mobile sensing? In *Proceedings of the 16th International Workshop on Mobile Computing Systems and Applications*, HotMobile '15, pages 117–122, New York, NY, USA, 2015. ACM.

[62] N. D. Lane, M. Mohammod, M. Lin, X. Yang, H. Lu, S. Ali, A. Doryab, E. Berke, T. Choudhury, and A. Campbell. Bewell: A smartphone application to monitor, model and promote wellbeing. In *5th international ICST conference on pervasive computing technologies for healthcare*, pages 23–26, 2011.

[63] N. D. Lane, Y. Xu, H. Lu, S. Hu, T. Choudhury, A. T. Campbell, and F. Zhao. Enabling large-scale human activity inference on smartphones using community similarity networks (csn). In *Proceedings of the 13th International Conference on Ubiquitous Computing*, UbiComp '11, pages 355–364. ACM, 2011.

[64] Q. V. Le. Building high-level features using large scale unsupervised learning. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 8595–8598. IEEE, 2013.

[65] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[66] J. Lee, A. Banerjee, and S. K. Gupta. Mt-diet: Automated smartphone based diet assessment with infrared images. In *Pervasive Computing and Communications (PerCom), 2016 IEEE International Conference on*, pages 1–6. IEEE, 2016.

[67] S. Lee, C. Min, C. Yoo, and J. Song. Understanding customer malling behavior in an urban shopping mall using smartphones. In *Proceedings of the 2013 ACM Conference on Pervasive and Ubiquitous Computing Adjunct Publication*, UbiComp '13 Adjunct, 2013.

[68] Y. Lee, Y. Ju, C. Min, S. Kang, I. Hwang, and J. Song. Comon: Cooperative ambience monitoring platform with continuity and benefit awareness. In *Proceedings of the 10th International Conference on Mobile Systems, Applications, and Services*, MobiSys '12, pages 43–56, New York, NY, USA, 2012. ACM.

[69] Y. Lee, C. Min, C. Hwang, J. Lee, I. Hwang, Y. Ju, C. Yoo, M. Moon, U. Lee, and J. Song. Sociophone: Everyday face-to-face interaction monitoring platform using multi-phone sensor fusion. In *Proceeding of the 11th Annual International Conference on Mobile Systems, Applications, and Services*, MobiSys '13, pages 375–388, New York, NY, USA, 2013. ACM.

[70] J. Lester, T. Choudhury, and G. Borriello. A practical approach to recognizing physical activities. In K. P. Fishkin, B. Schiele, P. Nixon, and A. Quigley, editors, *Pervasive Computing: 4th International Conference, PERVASIVE 2006, Dublin, Ireland, May 7-10, 2006. Proceedings*, pages 1–16, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[71] M. Levy, B. A. Weitz, and D. Grewal. Retailing management, 1998.

[72] L. Liao, D. Fox, and H. Kautz. Extracting places and activities from gps traces using hierarchical conditional random fields. *Int. J. Rob. Res.*, 26(1):119–134, Jan. 2007.

[73] R. LiKamWa, Y. Liu, N. D. Lane, and L. Zhong. Moodscope: Building a mood sensor from smartphone usage patterns. In *Proceeding of the 11th Annual International Conference on Mobile Systems, Applications, and Services*, MobiSys '13, pages 389–402, New York, NY, USA, 2013. ACM.

[74] R. LiKamWa, B. Priyantha, M. Philipose, L. Zhong, and P. Bahl. Energy characterization and optimization of image sensing toward continuous mobile vision. In *Proceeding of the 11th annual international conference on Mobile systems, applications, and services*, pages 69–82, New York, NY, USA, 2013. ACM.

[75] C. L. Lisetti and F. Nasoz. Using noninvasive wearable computers to recognize human emotions from physiological signals. *EURASIP Journal on Advances in Signal Processing*, 2004(11):929414, 2004.

[76] J. Liu, E. Johns, L. Atallah, C. Pettitt, B. Lo, G. Frost, and G.-Z. Yang. An intelligent food-intake monitoring system using wearable sensors. In *Wearable and Implantable Body Sensor Networks (BSN), 2012 Ninth International Conference on*, pages 154–160. IEEE, 2012.

[77] J. Liu, B. Priyantha, T. Hart, H. S. Ramos, A. A. Loureiro, and Q. Wang. Energy efficient gps sensing with cloud offloading. In *Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems*, pages 85–98. ACM, 2012.

[78] L. Liu, C. Karatas, H. Li, S. Tan, M. Gruteser, J. Yang, Y. Chen, and R. P. Martin. Toward detection of unsafe driving with wearables. In *Proceedings of the 2015 Workshop on Wearable Systems and Applications*, WearSys '15, pages 27–32, New York, NY, USA, 2015. ACM.

[79] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[80] H. Lu, D. Frauendorfer, M. Rabbi, M. S. Mast, G. T. Chittaranjan, A. T. Campbell, D. Gatica-Perez, and T. Choudhury. StressSense: Detecting stress in unconstrained acoustic environments using smartphones. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, UbiComp '12, pages 351–360, New York, NY, USA, 2012. ACM.

[81] H. Lu, J. Yang, Z. Liu, N. D. Lane, T. Choudhury, and A. T. Campbell. The jigsaw continuous sensing engine for mobile phone applications. In *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, pages 71–84. ACM, 2010.

[82] S.-M. Mäkelä, S. Järvinen, T. Keränen, M. Lindholm, and E. Vildjiounaite. *Shopper behaviour analysis based on 3d situation awareness information*, pages 134–145. Springer, 2014.

[83] C. K. Martin, S. D. Anton, H. Walden, C. Arnett, F. L. Greenway, and D. A. Williamson. Slower eating rate reduces the food intake of men, but not women: Implications for behavioral weight control. *Behaviour Research and Therapy*, 45(10):2349 – 2359, 2007.

[84] S. Mazilu, U. Blanke, and G. Trster. Gait, wrist, and sensors: Detecting freezing of gait in parkinson's disease from wrist movement. In *2015 IEEE International Conference on Pervasive Computing and Communication Workshops (PerCom Workshops)*, March 2015.

[85] E. Miluzzo, C. T. Cornelius, A. Ramaswamy, T. Choudhury, Z. Liu, and A. T. Campbell. Darwin phones: The evolution of sensing and inference on mobile phones. In *Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services*, MobiSys '10, pages 5–20, New York, NY, USA, 2010. ACM.

[86] E. Miluzzo, N. D. Lane, K. Fodor, R. Peterson, H. Lu, M. Musolesi, S. B. Eisenman, X. Zheng, and A. T. Campbell. Sensing meets mobile social networks: The design, implementation and evaluation of the cenceme application. In *Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems*, SenSys '08, pages 337–350. ACM, 2008.

[87] J.-K. Min, A. Doryab, J. Wiese, S. Amini, J. Zimmerman, and J. I. Hong. Toss'n'turn: smartphone as sleep and sleep quality detector. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 477–486. ACM, 2014.

[88] M. Mirtchouk, C. Merck, and S. Kleinberg. Automated estimation of food type and amount consumed from body-worn audio and motion sensors. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '16, pages 451–462, New York, NY, USA, 2016. ACM.

[89] T. Mo, S. Sen, L. Lim, A. Misra, R. Balan, and Y. Lee. Cloud-based query evaluation for energy-efficient mobile sensing. In *Mobile Data Management (MDM), 2014 IEEE 15th International Conference on*, volume 1, pages 221–224, July 2014.

[90] W. W. Moe. Buying, searching, or browsing: Differentiating between online shoppers using in-store navigational clickstream. *Journal of Consumer Psychology*, 13(1):29 – 39, 2003.

[91] A. Möller, S. Diewald, L. Roalter, and M. Kranz. Mobimed: comparing object identification techniques on smartphones. In *Proceedings of the 7th Nordic Conference on Human-Computer Interaction: Making sense through design*, 2012.

[92] Monsoon power monitor. `https://www.msoon.com/LabEquipment/PowerMonitor/`. Online; Acccessed: 2017-03-27.

[93] M. E. Morris, Q. Kathawala, T. K. Leen, E. E. Gorenstein, F. Guilak, M. Labhard, and W. Deleeuw. Mobile therapy: case study evaluations of a cell phone application for emotional self-awareness. *Journal of medical Internet research*, 12(2), 2010.

[94] F. J. Mulhern and D. T. Padgett. The relationship between retail price promotions and regular price purchases. *Journal of Marketing*, 59(4):83–90, 1995.

[95] Myfitnesspal inc.: Free calorie counter, diet and exercise journal. `http://myfitnesspal.com/`. Online; Acccessed: 2017-03-27.

[96] V. Natale, M. Drejak, A. Erbacci, L. Tonetti, M. Fabbri, and M. Martoni. Monitoring sleep with a smartphone accelerometer. *Sleep and Biological Rhythms*, 10(4):287–292, 2012.

[97] S. Nath. Ace: Exploiting correlation for energy-efficient and continuous context sensing. In *Proceedings of the 10th International Conference on Mobile Systems, Applications, and Services*, MobiSys 2012, 2012.

[98] S. Nishiguchi, M. Yamada, K. Nagai, S. Mori, Y. Kajiwara, T. Sonoda, K. Yoshimura, H. Yoshitomi, H. Ito, K. Okamoto, et al. Reliability and validity of gait analysis by android-based smartphone. *Telemedicine and e-Health*, 18(4):292–296, 2012.

[99] T. Okoshi, Y. Lu, C. Vig, Y. Lee, R. K. Balan, and A. Misra. Queuevadis: Queuing analytics using smartphones. In *Proceedings of the 14th International Conference on Information Processing in Sensor Networks*, pages 214–225. ACM, 2015.

[100] Omate truesmart. `https://www.omate.com`. Online; Acccessed: 2017-03-27.

[101] Open source computer vision library. `https://github.com/itseez/opencv`. Online; Acccessed: 2017-03-27.

[102] E. Owusu, J. Han, S. Das, A. Perrig, and J. Zhang. Accessory: Password inference using accelerometers on smartphones. In *Proceedings of the Twelfth Workshop on Mobile Computing Systems & Applications*, HotMobile '12, 2012.

[103] J. Paefgen, F. Kehr, Y. Zhai, and F. Michahelles. Driving behavior analysis with smartphones: insights from a controlled field study. In *Proceedings of the 11th International Conference on mobile and ubiquitous multimedia*, page 36. ACM, 2012.

[104] A. Parate, M.-C. Chiu, C. Chadowitz, D. Ganesan, and E. Kalogerakis. Risq: Recognizing smoking gestures with inertial sensors on a wristband. In *Proceedings of the 12th Annual International Conference on Mobile Systems, Applications, and Services*, MobiSys '14, pages 149–161, New York, NY, USA, 2014. ACM.

[105] T. Park, J. Lee, I. Hwang, C. Yoo, L. Nachman, and J. Song. E-gesture: A collaborative architecture for energy-efficient gesture recognition with hand-worn sensor and mobile devices. In *Proceedings of the 9th ACM Conference on Embedded Networked Sensor Systems*, SenSys '11, pages 260–273, New York, NY, USA, 2011. ACM.

[106] F. Pitta, T. Troosters, M. A. Spruit, V. S. Probst, M. Decramer, and R. Gosselink. Characteristics of physical activities in daily life in chronic obstructive pulmonary disease. *American journal of respiratory and critical care medicine*, 171(9):972–977, 2005.

[107] M. Popa, A. K. Koc, L. J. Rothkrantz, C. Shan, and P. Wiggers. Kinect sensing of shopping related actions. In *International Joint Conference on Ambient Intelligence*. Springer, 2011.

[108] M. Popa, L. Rothkrantz, C. Shan, T. Gritti, and P. Wiggers. Semantic assessment of shopping behavior using trajectories, shopping related actions, and context information. *Pattern Recognition Letters*, 34(7):809 – 819, 2013. Scene Understanding and Behaviour Analysis.

[109] M. Popa, L. Rothkrantz, Z. Yang, P. Wiggers, R. Braspenning, and C. Shan. Analysis of shopping behavior based on surveillance system. In *Systems Man and Cybernetics (SMC), 2010 IEEE International Conference on*, pages 2512–2519. IEEE, 2010.

[110] S. Poppitt, D. Swann, A. Black, and A. Prentice. Assessment of selective under-reporting of food intake by both obese and non-obese women in a metabolic facility. *International Journal of Obesity & Related Metabolic Disorders*, 22(4), 1998.

[111] Power retail. `http://www.powerretail.com.au/multichannel/mobile-tracking-draws-consumer-disapproval/`. Online; Acccessed: 2017-04-27.

[112] B. Priyantha, D. Lymberopoulos, and J. Liu. Littlerock: Enabling energy-efficient continuous sensing on mobile phones. *IEEE Pervasive Computing*, 10(2):12–15, 2011.

[113] N. B. Priyantha. *The cricket indoor location system*. PhD thesis, Massachusetts Institute of Technology, 2005.

[114] R. Purta, S. Mattingly, L. Song, O. Lizardo, D. Hachen, C. Poellabauer, and A. Striegel. Experiences measuring sleep and physical activity patterns across a large college cohort with fitbits. In *Proceedings of the 2016 ACM International Symposium on Wearable Computers*, ISWC '16, pages 28–35, New York, NY, USA, 2016. ACM.

[115] J.-W. Qiu, C.-P. Lin, and Y.-C. Tseng. Ble-based collaborative indoor localization with adaptive multi-lateration and mobile encountering. In *Wireless Communications and Networking Conference (WCNC), 2016 IEEE*, pages 1–7. IEEE, 2016.

[116] K. K. Rachuri, C. Efstratiou, I. Leontiadis, C. Mascolo, and P. J. Rentfrow. Metis: Exploring mobile phone sensing offloading for efficiently supporting social sensing applications. In *Pervasive Computing and Communications (PerCom), 2013 IEEE International Conference on*, pages 85–93. IEEE, 2013.

[117] K. K. Rachuri, M. Musolesi, C. Mascolo, P. J. Rentfrow, C. Longworth, and A. Aucinas. Emotionsense: a mobile phones based adaptive platform for experimental social psychology research. In *Proceedings of the 12th ACM international conference on Ubiquitous computing*, pages 281–290. ACM, 2010.

[118] M. Radhakrishnan, S. Eswaran, A. Misra, D. Chander, and K. Dasgupta. Iris: Tapping wearable sensing to capture in-store retail insights on shoppers. In *Pervasive Computing and Communications (PerCom), 2016 IEEE International Conference on*, 2016.

[119] M. Radhakrishnan, A. Misra, R. K. Balan, and Y. Lee. Smartphones and ble services: Empirical insights. In *Mobile Ad Hoc and Sensor Systems (MASS), 2015 IEEE 12th International Conference on*, pages 226–234. IEEE, 2015.

[120] M. Radhakrishnan, S. Sen, V. Subbaraju, A. Misra, and R. Balan. Iot + small data: Transforming in-store shopping analytics & services. In *2016 Eighth International Conference on Communication Systems and Networks (COMSNETS)*, 2016.

[121] T. Rahman, A. T. Adams, M. Zhang, E. Cherry, B. Zhou, H. Peng, and T. Choudhury. Bodybeat: A mobile system for sensing non-speech body sounds. In *Proceedings of the 12th Annual International Conference on Mobile Systems, Applications, and Services*, MobiSys '14, pages 2–13, New York, NY, USA, 2014. ACM.

[122] S. Rallapalli, A. Ganesan, K. Chintalapudi, V. N. Padmanabhan, and L. Qiu. Enabling physical analytics in retail stores using smart glasses. In *Proceedings of the 20th Annual International Conference on Mobile Computing and Networking*, MobiCom '14, pages 115–126, New York, NY, USA, 2014. ACM.

[123] N. Ravi, N. Dandekar, P. Mysore, and M. L. Littman. Activity recognition from accelerometer data. In *Proceedings of the 17th Conference on Innovative Applications of Artificial Intelligence - Volume 3*, IAAI'05, pages 1541–1546. AAAI Press, 2005.

[124] R. Ravichandran, E. Saba, K.-Y. Chen, M. Goel, S. Gupta, and S. N. Patel. Wibreathe: Estimating respiration rate using wireless signals in natural settings in the home. In *Pervasive Computing and Communications (PerCom), 2015 IEEE International Conference on*, pages 131–139. IEEE, 2015.

[125] S. Reddy, A. Parker, J. Hyman, J. Burke, D. Estrin, and M. Hansen. Image browsing, processing, and clustering for participatory sensing: lessons from a dietsense prototype. In *Proceedings of the 4th workshop on Embedded networked sensors*, pages 13–17. ACM, 2007.

[126] J. Renfrew, K. D. Pettigrew, and S. I. Rapoport. Motor activity and sleep duration as a function of age in healthy men. *Physiology & Behavior*, 41(6):627 – 634, 1987.

[127] N. Roy, A. Misra, and D. Cook. Infrastructure-assisted smartphone-based adl recognition in multi-inhabitant smart environments. In *Pervasive Computing and Communications (PerCom), 2013 IEEE International Conference on*, pages 38–46. IEEE, 2013.

[128] E. Sazonov, S. Schuckers, P. Lopez-Meyer, O. Makeyev, N. Sazonova, E. L. Melanson, and M. Neuman. Non-invasive monitoring of chewing and swallowing for objective quantification of ingestive behavior. *Physiological measurement*, 29(5):525, 2008.

[129] D. Schwarz, M. Schwarz, J. Stückler, and S. Behnke. Cosero, find my keys! object localization and retrieval using bluetooth low energy tags. In *Robot Soccer World Cup*, pages 195–206. Springer, 2014.

[130] S. Sen, D. Chakraborty, V. Subbaraju, D. Banerjee, A. Misra, N. Banerjee, and S. Mittal. Accommodating user diversity for in-store shopping behavior recognition. In *Proceedings of the 2014 ACM International Symposium on Wearable Computers*, pages 11–14, 2014.

[131] S. Sen, K. Grover, V. Subbaraju, and A. Misra. Inferring smartphone keypress via smartwatch inertial sensing. In *Pervasive Computing and Communication Workshops (PerCom Workshops), 2017 IEEE International Conference on*. IEEE, 2017.

[132] S. Sen, V. Subbaraju, A. Misra, R. Balan, and Y. Lee. Experiences in building a real-world eating recogniser. In *Proc. of WPA'17*, 2017.

[133] S. Sen, V. Subbaraju, A. Misra, R. K. Balan, and Y. Lee. The case for smartwatch-based diet monitoring. In *Pervasive Computing and Communication Workshops (PerCom Workshops), 2015 IEEE International Conference on*, pages 585–590. IEEE, 2015.

[134] S. Sen, V. Subbaraju, A. Misra, Y. Lee, and R. K. Balan. Demo: Smartwatch based food diary & eating analytics. In *Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services Companion*, pages 118–118. ACM, 2016.

[135] L. Shangguan, Z. Zhou, X. Zheng, L. Yang, Y. Liu, and J. Han. Shopminer: Mining customer shopping behavior in physical clothing stores with cots rfid devices. In *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems*, SenSys '15, pages 113–125, New York, NY, USA, 2015. ACM.

[136] L. K. Simone, N. Sundarrajan, X. Luo, Y. Jia, and D. G. Kamper. A low cost instrumented glove for extended monitoring and functional hand assessment. *Journal of neuroscience methods*, 160(2):335–348, 2007.

[137] Smart Nation Singapore. `https://www.smartnation.sg/`. Online; Acccessed: 2017-03-27.

[138] M. Soleymani, S. Asghari-Esfeden, M. Pantic, and Y. Fu. Continuous emotion detection using eeg signals and facial expressions. In *Multimedia and Expo (ICME), 2014 IEEE International Conference on*, pages 1–6. IEEE, 2014.

[139] R. Sommer, M. Wynes, and G. Brinkley. Social facilitation effects in shopping behavior. *Environment and Behavior*, 24(3):285–297, 1992.

[140] M. Stäger, P. Lukowicz, and G. Tröster. Implementation and evaluation of a low-power sound-based user activity recognition system. In *Wearable Computers, 2004. ISWC 2004. Eighth International Symposium on*, volume 1, pages 138–141. IEEE, 2004.

[141] Statista. `https://www.statista.com/statistics/461548/wearable-tech-sales-worldwide-by-category/`. Online; Acccessed: 2017-03-27.

[142] O. Storz, A. Friday, N. Davies, J. Finney, C. Sas, and J. Sheridan. Public ubiquitous computing systems: Lessons from the e-campus display deployments. *IEEE Pervasive Computing*, 5(3):40–47, July 2006.

[143] E. M. Tapia, S. S. Intille, and K. Larson. Activity recognition in the home using simple and ubiquitous sensors. In *International Conference on Pervasive Computing*, pages 158–175. Springer, 2004.

[144] E. M. Tauber. Why do people shop? *The Journal of Marketing*, pages 46–49, 1972.

[145] E. Thomaz, I. Essa, and G. D. Abowd. A practical approach for recognizing eating moments with wrist-mounted inertial sensing. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 1029–1040, New York, NY, USA, 2015. ACM.

[146] C. C. Tsai, G. Lee, F. Raab, G. J. Norman, T. Sohn, W. G. Griswold, and K. Patrick. Usability and feasibility of pmeb: a mobile phone application for monitoring real time caloric balance. *Mobile networks and applications*, 12(2-3):173–184, 2007.

[147] A. Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE transactions on Information Theory*, 13(2):260–269, 1967.

[148] C. Wang, X. Guo, Y. Wang, Y. Chen, and B. Liu. Friend or foe?: Your wearable devices reveal your personal pin. In *Proceedings of the 11th ACM on Asia Conference on Computer and Communications Security*, ASIA CCS '16, pages 189–200, New York, NY, USA, 2016. ACM.

[149] H. Wang, T. T.-T. Lai, and R. Roy Choudhury. Mole: Motion leaks through smartwatch sensors. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, MobiCom '15, pages 155–166, 2015.

[150] R. Wang, F. Chen, Z. Chen, T. Li, G. Harari, S. Tignor, X. Zhou, D. Ben-Zeev, and A. T. Campbell. Studentlife: Assessing mental health, academic performance and behavioral trends of college students using smartphones. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '14, pages 3–14, New York, NY, USA, 2014. ACM.

[151] Y. Wang, J. Lin, M. Annavaram, Q. A. Jacobson, J. Hong, B. Krishnamachari, and N. Sadeh. A framework of energy efficient mobile sensing for automatic user state recognition. In *Proceedings of the 7th International Conference on Mobile Systems, Applications, and Services*, MobiSys '09, pages 179–192, New York, NY, USA, 2009. ACM.

[152] A. Ward, A. Jones, and A. Hopper. A new location technique for the active office. *IEEE Personal Communications*, 4(5):42–47, Oct 1997.

[153] B. Wei, W. Hu, M. Yang, and C. T. Chou. Radio-based device-free activity recognition with radio frequency interference. In *Proceedings of the 14th International Conference on Information Processing in Sensor Networks*, IPSN '15, pages 154–165, New York, NY, USA, 2015. ACM.

[154] W. D. Wells and L. A. Lo Sciuto. Direct observation of purchasing behavior. *Journal of Marketing Research*, pages 227–233, 1966.

[155] S. Wesley, M. LeHew, and A. G. Woodside. Consumer decision-making styles and mall shopping behavior: Building theory using exploratory data analysis and the comparative method. *Journal of Business Research*, 59(5):535–548, 2006.

[156] C. R. Wren, A. Azarbayejani, T. Darrell, and A. P. Pentland. Pfinder: Real-time tracking of the human body. *IEEE Transactions on pattern analysis and machine intelligence*, 19(7):780–785, 1997.

[157] C. R. Wren and E. M. Tapia. Toward scalable activity recognition for sensor networks. In *International Symposium on Location-and Context-Awareness*, pages 168–185. Springer, 2006.

[158] D. Wyatt, M. Philipose, and T. Choudhury. Unsupervised activity recognition using automatically mined common sense. In *AAAI*, volume 5, pages 21–27, 2005.

[159] Z. Yan, V. Subbaraju, D. Chakraborty, A. Misra, and K. Aberer. Energy-efficient continuous activity recognition on mobile phones: An activity-adaptive approach. In *Wearable Computers (ISWC), 2012 16th International Symposium on*, pages 17–24. Ieee, 2012.

[160] K. Yatani and K. N. Truong. Bodyscope: a wearable acoustic sensor for activity recognition. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, pages 341–350, New York, NY, USA, 2012. ACM.

[161] X. Ye, G. Chen, and Y. Cao. Automatic eating detection using head-mount and wrist-worn accelerometers. In *E-health Networking, Application & Services (HealthCom), 2015 17th International Conference on*, pages 578–581. IEEE, 2015.

[162] C. W. You, C. C. Wei, Y. L. Chen, H. h. Chu, and M. S. Chen. Using mobile phones to monitor shopping time at physical stores. *IEEE Pervasive Computing*, 10(2):37–43, April 2011.

[163] M. Youssef and A. Agrawala. The horus wlan location determination system. In *Proceedings of the 3rd international conference on Mobile systems, applications, and services*, pages 205–218. ACM, 2005.

[164] Y. Zeng, P. H. Pathak, P. Mohapatra, C. Xu, A. Pande, A. Das, S. Miyamoto, E. Seto, E. Henricson, J. Han, et al. Analyzing shoppers behavior through wifi signals. In *Proc. of WPA'15*, 2015.

[165] X. Zhang, S. Li, R. R. Burke, and A. Leykin. An examination of social influence on shopper behavior using video tracking data. *Journal of Marketing*, 78(5):24–41, 2014.

[166] B. Zhou, J. Cheng, M. Sundholm, A. Reiss, W. Huang, O. Amft, and P. Lukowicz. Smart table surface: A novel approach to pervasive dining monitoring. In *Pervasive Computing and Communications (PerCom), 2015 IEEE International Conference on*, pages 155–162. IEEE, 2015.

[167] B. Zhou, M. Sundholm, J. Cheng, H. Cruz, and P. Lukowicz. Never skip leg day: A novel wearable approach to monitoring gym leg exercises. In *2016 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 1–9, March 2016.

[168] F. Zhu, M. Bosch, C. J. Boushey, and E. J. Delp. An image analysis system for dietary assessment and evaluation. In *Image Processing (ICIP), 2010 17th IEEE International Conference on*, pages 1853–1856. IEEE, 2010.